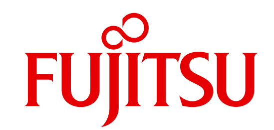
Deutsch



BS2000/OSD

# BS2000/OSD-BC V9.0

## Einführung in die Systembetreuung

Benutzerhandbuch

Ausgabe Juni 2013

## Kritik… Anregungen… Korrekturen…

Die Redaktion ist interessiert an Ihren Kommentaren zu diesem Handbuch. Ihre Rückmeldungen helfen uns, die Dokumentation zu optimieren und auf Ihre Wünsche und Bedürfnisse abzustimmen.

Sie können uns Ihre Kommentare per E-Mail an manuals@ts.fujitsu.com senden.

## Zertifizierte Dokumentation nach DIN EN ISO 9001:2008

Um eine gleichbleibend hohe Qualität und Anwenderfreundlichkeit zu gewährleisten, wurde diese Dokumentation nach den Vorgaben eines Qualitätsmanagementsystems erstellt, welches die Forderungen der DIN EN ISO 9001:2008 erfüllt.

cognitas. Gesellschaft für Technik-Dokumentation mbH
www.cognitas.de

## Copyright und Handelsmarken

# Inhalt

# 1 Einleitung

## 1.1 Zielgruppen und Zielsetzung des Handbuchs

Das Handbuch „Einführung in die Systembetreuung“ wendet sich an die Systemverwaltung und das Operating des Betriebssystems BS2000/OSD.
Es soll helfen, das Betriebssystem zu verwalten, zu steuern und zu überwachen.

## 1.2 Konzept des Handbuchs

Das Handbuch „Einführung in die Systembetreuung“ informiert über Themen zur Verwaltung und Überwachung von BS2000/OSD.

Folgende Themen werden nicht in diesem Handbuch, sondern in eigenen Handbüchern beschrieben:

- „SDF (System Dialog Facility)“,
  siehe die Handbücher „Dialogschnittstelle SDF“ [45] und „SDF-A“ [46].
- „Verwaltung von Subsystemen“,
  siehe Handbuch „Verwaltung von Subsystemen (DSSM/SSCM)“ [18].
- „Produktlieferung und Produktinstallation“, siehe Handbuch „IMON“ [25].
- „Erstellung und Analyse von Diagnoseunterlagen“,
  „Online-Wartung (WARTOPT)“,
  „Fehler- und Protokolldateien (HEL, SERSLOG, CONSLOG, RESLOG),
  siehe das „Diagnosehandbuch“ [14].

Die in diesem Handbuch genannten Kommandos bzw. Makros werden in den Handbüchern „Kommandos“ [27] bzw. „Makroaufrufe an den Ablaufteil“ [30] und „DVS-Makros“ [20] beschrieben, wenn nicht auf ein anderes Handbuch verwiesen wird.

Am Ende des Handbuchs finden Sie einen Anhang und verschiedene Verzeichnisse, die Ihnen das Arbeiten mit diesem Handbuch erleichtern.

### Readme-Datei

Funktionelle Änderungen der aktuellen Produktversion und Nachträge zu diesem Handbuch entnehmen Sie bitte ggf. der produktspezifischen Readme-Datei.

Readme-Dateien stehen Ihnen online bei dem jeweiligen Produkt zusätzlich zu den Produkthandbüchern unter *http://manuals.ts.fujitsu.com* zur Verfügung. Alternativ finden Sie Readme-Dateien auch auf der Softbook-DVD.

*Informationen unter BS2000/OSD*

Wenn für eine Produktversion eine Readme-Datei existiert, finden Sie im BS2000-System die folgende Datei:

`SYSRME.<product>.<version>.<lang>`

Diese Datei enthält eine kurze Information zur Readme-Datei in deutscher oder englischer Sprache (<lang>=D/E). Die Information können Sie am Bildschirm mit dem Kommando `/SHOW-FILE` oder mit einem Editor ansehen.
Das Kommando `/SHOW-INSTALLATION-PATH INSTALLATION-UNIT=<product>` zeigt, unter welcher Benutzerkennung die Dateien des Produkts abgelegt sind.

*Ergänzende Produkt-Informationen*

Aktuelle Informationen, Versions-, Hardware-Abhängigkeiten und Hinweise für Installation und Einsatz einer Produktversion enthält die zugehörige Freigabemitteilung. Solche Freigabemitteilungen finden Sie online unter *http://manuals.ts.fujitsu.com*.

## 1.3 Änderungen gegenüber dem Vorgänger-Handbuch

Das vorliegende Handbuch enthält gegenüber dem Vorgänger-Handbuch die nachfolgenden wesentlichen Änderungen.

Die Neuerungen von BS2000/OSD V9.0 finden Sie in den entsprechenden Vertriebsschriften und in der Freigabemitteilung von BS2000/OSD-BC (Produkt `BS2CP`, Version `180`) unter *http://manuals.ts.fujitsu.com*.

Alle Kapitel

- Anpassung an BS2000/OSD V9.0.
- SX-Server werden nicht mehr bedient. Ihre Beschreibung wurde gestrichen.
- Der Name „CentricStor“ wird nicht mehr verwendet. Die Produktbezeichnung und das entsprechende Benutzerhandbuch lauten ETERNUS CS HE.
- Bedienen des Net-Storage als externes Speichermedium.

Kapitel „Parameterservice“

- Neuer Parameter EPOCH im Parametersatz GTIME.

Kapitel „Geräteverwaltung“

- Die MBK-Systeme 3591 werden nicht mehr bedient. Ihre Beschreibung wurde gestrichen.
- Die neue Funktion XPAV wird beschreiben.

Neues Kapitel „Net-Storage-Verwaltung“.

Kapitel „Systemzeit-Verwaltung“

- Neuer Abschnitt „TODR-Epochen“.

Kapitel „Anhang“

- Neue Systemparameter: DIATTL=Q, SYSGJASL.

## 1.4 Darstellungsmittel

Wegen der häufigen Nennung der Bezeichnungen, werden der Einfachheit und Übersichtlichkeit halber folgende Abkürzungen gebraucht:

- **S-Server** für die Business Server der S-Serie (/390-Architektur)
- **SQ-Server** für die Business Server der SQ-Serie (X86-64-Architektur)
- **BS2000** für das Betriebssystem BS2000/OSD
- **Systemparameter** für die Klasse-2-Systemparameter von BS2000/OSD-BC

Die Zeichenfolgen `<date>`, `<time>` und `<version>` bezeichnen in Beispielen die aktuellen Ausgaben für Datum, Uhrzeit und Version eines Software-Produktes, wenn die Beispiele sonst Datums-, Zeit- und Versions-unabhängig sind.

In diesem Handbuch werden folgende Darstellungsmittel verwendet:

**`Eingabe`** Eingaben in Beispielen werden in halbfetter Schreibmaschinenschrift dargestellt

`Ausgabe` DSECTS, Übersetzungslisten oder Ausgaben in Beispielen werden in Schreibmaschinenschrift dargestellt

i Dieses Zeichen kennzeichnet Hinweise auf wichtige Informationen

! Dieses Zeichen kennzeichnet einen Warnhinweis, der auf die Möglichkeit des Datenverlustes oder anderer ernsthafter Schäden an Daten hinweist.

[ ] Literaturhinweise werden im Text in Kurztiteln angegeben. Der vollständige Titel jeder Druckschrift, auf die durch eine Nummer verwiesen wird, ist im Literaturverzeichnis hinter der entsprechenden Nummer aufgeführt.

## 1.5 Aufgaben der Systemverwaltung

Die Administration von BS2000/OSD umfasst diejenigen Aufgaben, welche im Hinblick auf eine effiziente, sichere und optimale Nutzung des Systems wahrgenommen werden müssen.

Die Aufgaben der Systemverwaltung sind:

- Zusammenstellen und Einführen der zweckmäßigen Betriebssystemkomponenten
- Verwalten und Aktualisieren des im Data Center eingesetzten Organisationsprogramms und der Programmbibliotheken
- Sicherstellen der Funktionsfähigkeit der System- und systemnahen Software
- Verwalten der Zugangsberechtigung zum System
- Verwalten und Verteilen der Betriebsmittel
- Benutzer- und Dateienverwaltung
- Auswerten von Fehlerstatistiken
- Sichern der System- und Benutzerdaten
- Installieren von Software-Produkten
- Ergreifen von Maßnahmen zur Rationalisierung und Automatisierung des Betriebes im Data Center.
- Einrichten und Zuordnen von Operator-Kennungen und Operator-Rollen (im Zusammenhang mit der Privilegienverwaltung).
- Erstellung von Diagnoseunterlagen bei Betriebssystem-, Benutzerprogramm- und Bedienungsfehlern

Der Systemverwaltung stehen mit privilegierten Benutzerkennungen auf Kommando- und Dienstprogrammebene geeignete Werkzeuge für diese Aufgaben zur Verfügung.

#### Kommandos

Neben speziellen Systembetreuerkommandos, die für den privilegierten Aufrufer reserviert sind, können sowohl sämtliche Benutzerkommandos als auch eine Untermenge der Operatorkommandos eingesetzt werden.
Die Untermenge an Operatorkommandos erlaubt der Systemverwaltung, an der Bedienschnittstelle steuernden und überwachenden Einfluss auf Domänen der Systembedienung, wie z.B. Lastverteilung und Auftragssteuerung, auszuüben.

Bei den Benutzerkommandos wirkt sich die Privilegierung z.B. der Benutzerkennung TSOS in einem erweiterten Funktionsumfang aus, der der Systemverwaltung angeboten wird. Diese Funktionen realisieren im Wesentlichen eine präzisere, umfassendere Informations-

ausgabe und die für den „Super-User“ eines Betriebssystems typische Aufhebung von Betriebsmittel-, Prozess- und Kataloggrenzen, die dem nicht-privilegierten Benutzer gesetzt sind.

**Dienstprogramme**

Der Systemverwaltung stehen sämtliche Dienstprogramme, die auch die Benutzer anwenden dürfen, mit einem zum Teil erweiterten Funktionsumfang zur Verfügung (z.B. DPAGE, ARCHIVE, JMU, SPCCNTRL).

Darüber hinaus kann sich die Systemverwaltung spezieller Dienstprogramme bedienen, die nur unter einer privilegierten Benutzerkennung laufen und zur Steuerung und Überwachung des Betriebssystems eingesetzt werden.

## 1.6 Aufgaben des Operating

Dem Operator obliegen die Steuerung der Inbetriebnahme des Systems, Eingriffe zur Regelung des Systems, die Kommunikation mit dem Benutzer und die Bedienung der peripheren Geräte. Systeminitialisierung (Startup) und Systemspeicherabzug (SLED) können ausschließlich vom Operator bedient werden. Für Funktionen zur Hardware-Diagnose und Rekonfiguration, die dem Operator an der Bedienstation vorbehalten sind, stehen ihm weitere SVP-Kommandos zur Verfügung.

Für die übrigen Aufgaben verwendet der Operator Kommandos (spezielle Operatorkommandos bzw. Benutzer- oder Systemverwaltungskommandos), deren Nutzung im vorliegenden Handbuch ebenfalls beschrieben ist. Zu den wesentlichen Aufgaben des Operators zählen u.a.:

- Starten und Überwachen des Datenkommunikationssystems
- Rekonfigurieren der Hardwarekomponenten und der Verbindungen zur Peripherie
- Bereitstellen und Zuweisen externer Datenträger
- Überwachen und Steuern der Gerätebelegung
- Steuerung der Betriebsmittelzuweisung
- Betriebsüberwachung
- Kommunikation mit den Benutzern

## 1.7 Automatisierung

Der Haupteinsatzbereich von BS2000-Systemen ist der performante und störungsfreie Ablauf von Kundenanwendungen.
Bei vielen BS2000-Kunden ist der Betrieb vollständig automatisiert, d.h. die folgenden Phasen laufen vollautomatisch und zuverlässig ab:

- Inbetriebnahme (vom Systemstart bis zur Verfügbarkeit der Anwendungen)
- Betrieb (Systempflege und Ablauf der Anwendungen)
- Systembeendigung

Durch den Einsatz moderner Hardware, wie robotergestützte Magnetbandkassetten-Systeme, kann die manuelle Bedienung von Geräten weitgehend entfallen.
Aus Sicht der Software ist es notwendig, den Ablauf dieser Phasen an die Hardware- und Software-Gegebenheiten des Kunden anzupassen („Customizing“).

Mit der Kommandosprache SDF-P und dem Kommunikationsmittel Jobvariable (beides kostenpflichtige Produkte; außer SDF-P-BASYS, dem Basisteil von SDF-P) können die genannten Abläufe auf Kommandoebene programmiert/automatisiert werden.

In den Phasen der Inbetriebnahmne und Systembeendigung, in denen diese Werkzeuge noch nicht oder nicht mehr zur Verfügung stehen, werden die datei- oder datenorientierten Schnittstellen des SVP (z.B. Ein-/Ausschaltsequenzen), der Systemeinleitung (Startup-Parameterservice) und des DSSM (Subsystemkatalog) angeboten.

Für alle weitergehenden Automatisierungsaufgaben der Systembetreuung, wie z.B. die automatische Reaktion auf Ereignisse, bietet das Software-Produkt PROP-XT passende Benutzerkommandos an und stellt die zugehörigen Daten in S-Variablen bereit.

So steht – durch die ergänzende Funktion von PROP-XT – mit SDF-P und Jobvariablen ein komfortables, einheitliches Werkzeug zur Automatisierung aller Aufgaben der System-betreuung zur Verfügung.

# 2 Systemeinleitung und -beendigung

Dieses Kapitel beschreibt die Abläufe bei der Systemeinleitung und -beendigung.

## 2.1 Überblick über die Abläufe bei der Systemeinleitung

Die Systemeinleitung von BS2000/OSD läuft als „Bootstrapping“ ab, d.h. es werden schrittweise immer mächtigere Funktionseinheiten geladen und gestartet, bis BS2000/OSD betriebsbereit ist.

Der Anstoß zum Ablauf der verschiedenen Routinen erfolgt hardware-abhängig über den Serviceprozessor (SVP, S-Server) oder X2000 (SQ-Server) bzw. die Restart-Verarbeitung bei automatischem Restart. Mit diesem Urladen (IPL, Initial Program Load) wird die Systemeinleitung gestartet. Dabei wird neben der IPL-Platte auch die Art der Systemeinleitung festgelegt. Durch die Einstellung der Ladeoptionen für BS2000/OSD wird festgelegt, ob der Systemstart komfortabel oder flexibel ablaufen soll.

Für einen komfortablen, weitgehend automatischen Ablauf stehen dem Operator die Modi FAST und AUTOMATIC, für den flexiblen, dialogorientierten Ablauf steht ihm der Modus DIALOG zur Verfügung (siehe Abschnitt „Arten der Systemeinleitung“ auf Seite 44).

Für ein VM2000-Gastsystem wird die Systemeinleitung durch das VM2000-Kommando START-VM (siehe Handbuch „VM2000“ [62]) ausgelöst. Auf SQ-Servern kann das VM2000-Gastsystem auch über den SQ-Manager oder über die SVP-Funktionen der KVP-Konsole, die der virtuellen Maschine zugeordnet ist, gestartet werden.

Die wichtigsten Funktionsabläufe während der Systemeinleitung von der Bereitstellung der Hardware bis zum eigentlichen Abschluss des Startups, bis „System Ready“ sind:

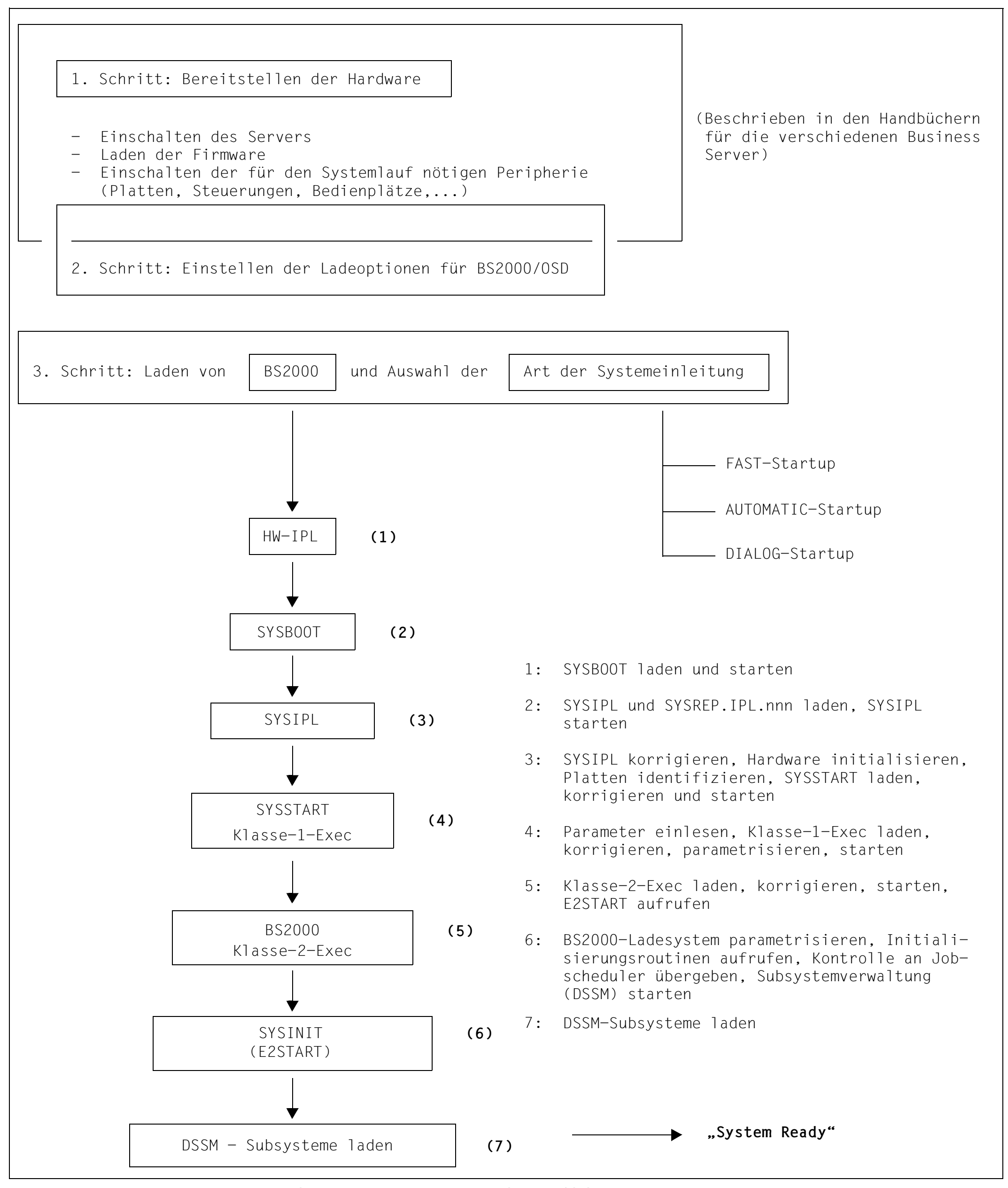



Bild 1: Funktionsabläufe bei der Systemeinleitung von BS2000/OSD

Die Systemeinleitung für BS2000/OSD kann beginnen, wenn die dazu notwendige Hardware-Einheiten (Server, lokale Bedienstationen, periphere Geräte) eingeschaltet und betriebsbereit sind. Der Ablauf dieser Schritte (Einschalten der Stromversorgung, Laden der Firmware usw.) ist ausführlich in den Handbüchern für den jeweiligen BS2000/OSD-Server beschrieben.

Der interne Ablauf der Systemeinleitung für BS2000/OSD beginnt mit dem Laden des sog. „Urladers“ (SYSBOOT). Dies ist die eigentliche Aufgabe der Routine HW-IPL (Vorgang (1) in Bild 1).

SYSBOOT ist das erste Programm der Systemeinleitung, das elementare Prüfungen vornimmt und eine weitere Laderoutine anstößt (Vorgang (2) in Bild 1).

Diese von SYSBOOT geladene und gestartete Routine ist SYSIPL, die die Aufgabe hat, die Optionen im Modus DIALOG abzufragen und die aktuelle Platten- und Prozessorkonfiguration zu ermitteln (Vorgang (3) in Bild 1). Die Zeitbasis der Systemzeit wird festgelegt. Die Plattenkonfiguration wird auf Vollständigkeit und Eindeutigkeit geprüft. Bei DRV-Platten im Home-Pubset werden die zusammengehörigen Plattenpaare ermittelt. Außerdem wird von dieser Routine entweder SYSSTART oder SLED geladen und jeweils korrigiert.

Die beiden Programme SYSBOOT und SYSIPL und die IPL-Rep-Datei befinden sich auf festen Plätzen eines bestimmten Plattenspeichers, der IPL-Platte. Die IPL-Platte kann ein gemeinschaftlicher Plattenspeicher – die Platte eines Pubsets – oder eine Privatplatte sein. Sie muss für FAST- und AUTOMATIC-Startup eine der Platten des Home-Pubsets sein.



Um im Fall eines Systemabsturzes den automatischen Restart zu ermöglichen, sollte bei DRV-Pubsets immer die Platte mit der niedrigeren Subchannel-Number als IPL-Platte angegeben werden.

Nur bei DIALOG-Startup ist ein IPL von Privatplatte möglich. Dabei muss der Operator im späteren Verlauf der Systemeinleitung angeben, welcher Pubset der Home-Pubset für den Systemlauf sein soll.

Beim IPL von gemeinschaftlicher Platte wird der Pubset, zu dem die Platte gehört, automatisch zum Home-Pubset gewählt. Nur bei Dialog-Startup mit der Option ALLDISK kann der Operator den Pubset noch wechseln.

Wenn die IPL-Platte nicht zum späteren Home-Pubset gehört, ist insbesondere darauf zu achten, dass Versionen und Korrekturstände auf beiden gleich sind.



In einer Plattenkonfiguration können durchaus mehrere IPL-Platten existieren.
Das Einrichten von IPL-Platten ist Aufgabe der Systembetreuung und wird mit Hilfe des Dienstprogramms SIR vorgenommen, das ausführlich im Handbuch „Dienstprogramme“ [15] beschrieben ist.
Eine IPL-Platte kann entweder für S-Server oder für SQ-Server verwendet werden.

Beim Laden von SYSBOOT und SYSIPL/SLED stehen die Dateiverwaltungs-Funktionen von BS2000/OSD noch nicht zur Verfügung. Die notwendigen Dateien können daher nur gefunden werden, wenn sie zuvor auf der IPL-Platte „verankert“ worden sind. Dies geschieht mit der Anweisung CREATE-IPL-VOLUME des Dienstprogramms SIR und beinhaltet folgende Einzelschritte:

- Die zum Laden von SYSBOOT und SYSIPL/SLED nötigen Dateien werden auf die IPL-Platte kopiert.
- Die von SYSBOOT und SLED verwendeten Sicherstellungsdateien werden auf der IPL-Platte erzeugt.
- Im SVL der IPL-Platte wird je ein unmittelbarer Verweis auf jede dieser Dateien eingetragen:

| Original-Datei | von SIR erzeugte Datei |
|---|---|
| --- | SYSDAT.IPL-CONF.DSKnnn |
| --- | SYSPRG.BOOT.DSKnnn.SAVE |
| SYSPRG.IPL.180 | SYSPRG.IPL.DSKnnn |
| --- | SYSPRG.SLED.DSKnnn.SAVE |
| SYSREP.IPL.180 | SYSREP.IPL.DSKnnn |
| SYSREP.SLED.180 | SYSREP.SLED.DSKnnn |

  „nnn“ steht jeweils für die Nummer der IPL-Platte innerhalb des Pubsets. Handelt es sich bei der IPL-Platte um eine Privatplatte, wird statt des Namensteils DSKnnn die VSN der Privatplatte eingesetzt.

- Sofern noch nicht auf einer anderen Platte des Pubsets vorhanden, wird die Sicherstellungsdatei für Systemkorrekturen SYS.NSI.SAVEREP erzeugt (nicht bei Privatplatten).

Die Original-Dateien werden zur Systemeinleitung nicht mehr gebraucht und können im laufenden System gefahrlos geändert werden, z.B. zur Übernahme eines neuen Korrekturstandes. Alle Änderungen werden für die Systemeinleitung jedoch erst wirksam, nachdem mit SIR neue Kopien erzeugt und verankert worden sind.

Die verankerten Dateien dürfen im laufenden System weder geändert noch gelöscht werden, weil dadurch i.A. die IPL-Fähigkeit der Platte zerstört wird. Sie werden seitens SIR durch BACKUP-CLASS=E und MIGRATE=INHIBITED vor Verlagerung und Verdrängung geschützt.

Alle weiteren für die Systemeinleitung benötigten Routinen befinden sich als „normale Dateien“ auf einem Pubset oder einer Privatplatte.
Die Konfigurationstabellen werden dynamisch erzeugt.

SYSSTART (Vorgang (4) in Bild 1) ist ein Programm, das die eigentliche Systemeinleitung für BS2000/OSD vorbereitet und durchführt. In der Vorbereitung werden i.W. die Parameter für BS2000/OSD eingelesen, die Objektcodekorrekturen für Klasse-1-Exec ermittelt, sowie die Versionsstände von SYSSTART und BS2000/OSD auf Konsistenz geprüft. In der Durchführung werden die einzelnen Initialisierungsfunktionen von BS2000/OSD tabellengesteuert aufgerufen. Zu diesen Initialisierungsfunktionen gehören auch die Datenstrukturen der virtuellen Speicherverwaltung und die Initialisierung der Paging-Bereiche, womit SYSSTART auch den Übergang in den virtuellen Adressierungsmodus von BS2000/OSD vorbereitet.

Von SYSSTART wird schließlich das BS2000-Ladesystem (Vorgang (5) in Bild 1), das aus den beiden Teilen „Klasse-1-Exec“ (resident) und „Klasse-2-Exec“ (seitenwechselbar) besteht, aufgerufen.
Auf dieser Stufe der Systemeinleitung werden u.a. bei der Geräteverwaltung Zugriffsrechte für die Geräte der Systemeinleitung beantragt (Platten des Home-Pubsets und der Paging-Pubsets). Nach Durchlauf dieser Systemeinleitungs-Phase liegt das BS2000-Ladesystem **geladen**, **korrigiert** und **parametrisiert** vor.

**Geladen** heißt, dass sich dieser Teil von BS2000/OSD nach dem Ladevorgang vollständig im Hauptspeicher befinden. Es werden zuerst die hauptspeicherresidenten Teile (Klasse-1-Exec) geladen. Die restlichen Teile sind seitenwechselbar. Diese Routinen werden von SYSINIT über den Hauptspeicher auf den Seitenwechselspeicher kopiert.

**Korrigiert** heißt, dass Module dieser Teile von SYSSTART während der Systemeinleitung mittels so genannter Rep-Sätze modifiziert werden. Die Rep-Sätze können von max. vier katalogisierten Plattendateien in beliebiger Reihenfolge eingelesen werden. Die Bedienstation kann – außer für Reps für SYSIPL – zusätzlich als Eingabegerät definiert werden. Bei der Systemeinleitung werden alle verarbeiteten Rep-Sätze von Platte und Bedienstation in die Sicherstellungsdatei SYS.NSI.SAVEREP geschrieben und später in der Datei $SYSAUDIT.SYS.REPLOG.<datum>.<session-nr>.01 protokolliert.

**Parametrisiert** heißt, dass eine Menge von Parametersätzen, die Anweisungen für die Initialisierungsroutinen von BS2000/OSD enthalten, eingelesen werden. Die gesamte Parametereingabe besteht aus einer Folge von Abschnitten, die – durch spezifische Schlüsselwörter gekennzeichnet – jeweils bestimmte Funktionseinheiten betreffen und von diesen ausgewertet werden (siehe Kapitel „Parameterservice“ auf Seite 75).

Der Operator kann durch eine Voreinstellung beim Systemstart entscheiden, ob das Laden, Korrigieren und Parametrisieren weitestgehend automatisch und dialogfrei ablaufen soll (in diesem Fall werden jeweils Dateien mit Standardnamen herangezogen), oder ob der Ablauf flexibel – im Dialog mit dem Operator – gesteuert werden soll.

Abschließend wird mit E2START die letzte Phase der Systemeinleitung für BS2000/OSD gestartet (Vorgang (6) in Bild 1). Diese Routine läuft bereits unter BS2000/OSD und bestimmt zunächst den Namen der Kommandodatei (CMDFILE), die nach „System Ready“ automatisch angestartet werden soll.

Zur Erreichung eines funktionstüchtigen BS2000/OSD laufen in dieser Phase Initialisierungsroutinen für die folgenden Funktionseinheiten ab:

- Aktivierung des Task-Schedulers
- Öffnen der Systemdateien (Benutzerkatalog, SYSEAM, usw.)
- Bereitstellung der Dateikatalogverwaltung
- Aktivierung des dynamischen Bindeladers (DBL)
- Aktivierung der Bibliothekszugriffsmethode PLAM
- Start der Funktionen zur Überwachung von Platten- und Bandgeräten
- Aktivierung der SERSLOG-Funktion
- Start der Dynamischen Subsystemverwaltung DSSM [1)]

Nach Freigabe des von der Systemeinleitung belegten Speichers und Start des Job-Schedulers ist „System Ready“ erreicht und die Abarbeitung der in der Kommandodatei CMDFILE hinterlegten Kommandos wird angestoßen. Zwar ist die Verwendung einer Kommandodatei – deren Name frei wählbar ist – nicht zwingend erforderlich, doch wegen der Forderung nach Automatisierung der Abläufe unbedingt zu empfehlen.

Die Verwendung einer Kommandodatei gewährleistet u.a. die automatische Aktivierung derjenigen Systemkomponenten und -einstellungen, die für die Funktionstüchtigkeit des spezifischen Systems relevant sind:

- Inbetriebnahme von optionalen Subsystemen
- Starten des BS2000-Datenkommunikationssystems [2)]
- Laden des SPOOL-Systems [3)]
- Spezifische Lastregulierung
- Aktivierung spezieller Programme über Enter-Dateien

*Hinweise*

1) DSSM:
Während der Systemeinleitung erhält die Dynamische Subsystemverwaltung die Steuerung. Diese initialisiert sich mit dem vorgegebenen Subsystemkatalog und aktiviert Subsysteme bzw. leitet die Aktivierung von Subsystemen ein. Der Startzeitpunkt eines Subsystems (vor oder nach „System Ready“) wird von der Systembetreuung bei der Deklaration festgelegt. Auf diese Weise können Subsysteme automatisch aktiviert werden.

Beim Start eines Subsystem ermittelt DSSM über IMON-GPN die Pfadnamen aller Dateien des Subsystems aus dem aktuellen SCI. Ist IMON-GPN nicht verfügbar (wird bereits beim Start von DSSM geladen) oder existiert keine Datei unter dem ermittelten Pfadnamen, verwendet DSSM die im Subsystemkatalog eingetragenen Standard-Namen. Bei Verwendung des Standard-Namens wird die Meldung `ESM0665` ausgegeben.

2) Datenkommunikationssystem (DCM):
Für den Start des Datenkommunikationssystems noch vor „System Ready“ kann das DCSTART-Kommando auch als BCAM-Parameter in der Startup-Parameterdatei vorgegeben werden (siehe Handbuch „BCAM“ [4], Abschnitt BCAM-BS2000-Parameterdatei).

Ist das nicht der Fall, muss nach jeder Systemeinleitung das Datenkommunikationssystem gesondert in Betrieb genommen werden. Dies geschieht mit dem Kommando DCSTART, das dann zweckmäßigerweise in der CMDFILE hinterlegt wird.

Mit dem Kommando DCSTART wird automatisch die Eröffnung der folgenden internen, privilegierten Anwendungen des Servers eingeleitet:

- $DIALOG (Anwendung für Dialogverarbeitung (TIAM))
- $CONSOLE (Anwendung für Universelle Konsole)
- $BCAM (Anwendung für den TRANSDATA DCM Informationsdienst)

Erfolgt das erste DCSTART-Kommando später als 10 Min. nach „System Ready“ oder wird bei laufendem BS2000/OSD das DCM beendet (Kommando BCEND) und neu gestartet, dann muss die Anwendung $DIALOG vom Operator mit dem Kommando START-DIALOG-APPLICATION manuell gestartet werden. Eine andere Möglichkeit wäre die Aufnahme von /START-DIALOG-APPLICATION in die SOF (Start Option File) von BCAM. Voraussetzung dafür ist die Einrichtung eines Konsolzugangs für BCAM mit der Berechtigung für START-DIALOG-APPLICATION (siehe Handbuch „BCAM“ [4]).

Weil im Bedienmodus mit Operator-Logon der Operator nach „System Ready“ erst ein SET-LOGON-PARAMETERS- und ein REQUEST-OPERATOR-ROLE-Kommando eingeben muss, um weitere Kommandos eingeben zu können, empfiehlt es sich, die Voraussetzungen für die ersten beiden Kommandos ebenfalls aus der CMDFILE heraus zu schaffen. Diese Voraussetzungen sind:

- die Operatorkennung muss entsperrt sein
- eine Operatorrolle muss eingerichtet sein
- die Operatorrolle muss der Operatorkennung zugeordnet sein

Da das Entsperren der Operatorkennung unter der Kennung TSOS geschehen muss, das Einrichten einer Operatorrolle und das Zuordnen zu einer Operatorkennung aber nur unter SYSPRIV durchgeführt werden kann, empfiehlt es sich, aus der CMDFILE einen ENTER-Job aufzurufen, der das UNLOCK-USER-Kommando absetzt und eine weitere Prozedur zum Einrichten und Zuordnen der Operatorrollen unter der Kennung SYSPRIV aufruft.

Der gesamte Ablauf könnte etwa folgendermaßen aussehen:

Aufruf aus der CMDFILE:

```
/ENTER-JOB E.OPR-LOGON.TSOS
```

*Inhalt der Prozedur E.OPR-LOGON.TSOS*

```
/SET-LOGON-PARAMETERS
/ UNLOCK-USER SYSPRIV
/ SET-JOB-STEP
/ UNLOCK-USER SYSOPR
/ SET-JOB-STEP
/ ENTER-JOB FROM-FILE=$TSOS.E.OPR-LOGON.SYSPRIV,-
/           PROC-ADMISS=*PAR(USER-ID=SYSPRIV,-
/                            ACC=SYSACC,-
/                            PASS=*NONE)
/EXIT-JOB
```

*Inhalt der Prozedur E.OPR-LOGON.SYSPRIV*

```
/SET-LOGON-PARAMETERS
/ CREATE-OPERATOR-ROLE OPERATOR-ROLE=SYSADM,ROUT-CODES=*ALL
/ SET-JOB-STEP
/ MODIFY-OPERATOR-ATTR USER-ID=SYSOPR,ADD-OP-ROLE=SYSADM
/ SET-JOB-STEP
/ INFORM-OPERATOR,-
/         MSG='*** OPERATOR-ROLE SYSADM CREATED AND ADDED ***'
/ INFORM-OPERATOR,-
/ MSG='+-----------------------------------------------------+'
/ INFORM-OPERATOR,-
/ MSG='!    THE FIRST OPERATOR COMMANDS AFTER SYSTEM READY    !'
/ INFORM-OPERATOR,-
/ MSG='!    (BEFORE /DCSTART ... )  MUST BE:                  !'
/ INFORM-OPERATOR,-
/ MSG='!    /SET-LOGON-PARAMETERS SYSOPR,SYSACC               !'
/ INFORM-OPERATOR,-
/ MSG='!    /REQUEST-OPERATOR-ROLE OP-ROLE=SYSADM             !'
/ INFORM-OPERATOR,-
/ MSG='+-----------------------------------------------------+'
/EXIT-JOB
```

Erst dann ist BS2000/OSD kommunikationsfähig.

3) SPOOL:
Nach jeder Systemeinleitung muss SPOOL gesondert geladen und initialisiert werden. Der SPOOL-Startup wird durch das Kommando START-SUBSYSTEM eingeleitet. Mit dem Operanden SUBSYSTEM-PARAMETER kann festgelegt werden, ob für SPOOL ein Warm- oder Kaltstart durchgeführt werden soll und ob zusätzlich das Software-Produkt RSO geladen werden soll. Dieses Kommando sollte entweder gleich nach „System Ready“ gegeben werden oder in der Kommandodatei CMDFILE enthalten sein. Ist SPOOL nicht geladen, können keine SPOOLIN- und SPOOLOUT-Jobs durchgeführt werden. SPOOL-Anforderungen des Operators (z.B. Kommandos START-PRINTER-OUPUT, MODIFY-PRINTER-OUTPUT-STATUS,...) werden entweder abgewiesen, ignoriert oder zurückgestellt.

Zusammenfassend besteht die Systemeinleitung aus folgenden **internen** Schritten:

HW-IPL:
- Laden des 1. Blocks von SYSBOOT

SYSBOOT:
- Laden des 2. Blocks von SYSBOOT
- Laden von SYSREP.IPL.<version>
- Laden und Starten von SYSIPL

SYSIPL:
- eigene Korrektur
- Laden, Korrigieren und Starten von SYSSTART

SYSSTART:
- Einlesen der Parameter
- Laden und Korrigieren der hauptspeicherresidenten Teile von BS2000/OSD (Klasse-1-Exec)
- Initialisierung des Seitenwechselspeichers

Klasse-1-Exec:
- Initialisierung des residenten Teils von BS2000/OSD
- Automatisches Zuschalten von Plattengeräten, die DETACHED generiert wurden und auf denen benötigte Public-Platten montiert sind [1]
- Laden des seitenwechselbaren Teils von BS2000/OSD
- (Klasse-2-Exec)
  Korrektur der seitenwechselbaren Teile

Klasse-2-Exec:
- Initialisierung der seitenwechselbaren Teile

SYSINIT (E2START):
- Ermittlung der Kommandodatei und Aufruf von Initalisierungsfunktionen für BS2000-Funktionseinheiten (DSSM, PLAM, usw.)
- Freigabe des belegten Speichers und Start des Jobschedulers

„System Ready“
(jetzt erfolgt z.B. der Anstoß der Kommandodatei CMDFILE)

[1] Während des Startups werden automatisch alle Public-Platten des Home-Pubsets und alle Pubsets, die Paging-Platten enthalten und die beim Parameterservice angegeben wurden, dem System zugeschaltet, auch wenn sie explizit DETACHED generiert worden sind. Diese Geräte bleiben während des gesamten Systemlaufs ATTACHED. Die Pfade zu diesen Geräten müssen jedoch INCLUDED generiert worden sein (siehe Seite 196).

Mit dem Kommando SHOW-SYSTEM-INFORMATION können Informationen über die Systemkonfiguration, die eingesetzte VM2000-Version, das Monitorsystem und die Parameter der Zeiteinstellung eingeholt werden.

### 2.1.1 Zeitbestimmung bei der Systemeinleitung

Siehe auch Kapitel „Systemzeit-Verwaltung“ auf Seite 705.

Für die Bestimmung von Datum und Uhrzeit während der Systemeinleitung existieren drei Quellen:

- SVP-Uhr: liefert die lokale Zeit
  Diese Uhr ist batteriegepuffert, sodass sie auch nach einem Netzausfall einen gültigen Wert liefert.
  - Auf S-Servern wird die SVP-Uhr durch den SKP 3970 synchronisiert. Die SKP-Uhr kann optional mit einem Zeitgeber synchronisiert sein, z.B. mit einem NTP-Server (NTP=Network Time Protocol).
  - Auf SQ-Servern wird der SVP von X2000 emuliert. Die SVP-Zeit entspricht der Zeit des Trägersystems. Die Trägersystem-Zeit kann optional mit einem Zeitgeber synchronisiert sein.
- CPU-Uhr (TODR): ergibt mit einem auf dem Home-Pubset hinterlegten Korrekturwert die lokale Zeit. Diese Uhr läuft auch bei CPU-Stop weiter, aber nicht bei einer Unterbrechung der Stromversorgung.
  - Auf S-Servern ist das TODR eine eigenständige Uhr.
  - Auf SQ-Servern wird das TODR von X2000 emuliert.
- Operator
  Dem Operator wird die aktuell gültige Zeit an der Bedienstation gezeigt. Bei Fehlersituationen oder im Modus DIALOG (mit der Option UNLOCK) werden ihm die entsprechenden Meldungen an der Bedienstation ausgegeben und er wird zur Bestätigung oder Korrektur aufgefordert.

Die lokale Zeit (LT) ist die am Installationsort von BS2000/OSD gültige gesetzliche Zeit. Durch gesetzlich vorgeschriebene Zeitumstellungen (Sommer-/Winterzeit) ist sie nicht monoton fortlaufend, sondern erfordert an den Umstellungszeitpunkten eine positive bzw. negative Korrektur. Diese Korrektur erfolgt automatisch mit Hilfe der GTIME-Parameterdatei, in der der Korrekturwert und die Umstellungszeitpunkte hinterlegt sind (siehe Seite 87).

Die Systemzeit basiert auf der lokalen Zeit. Sie wird bei Existenz eines externen Zeitgebers am SKP 3970 genau mit der gesetzlichen Zeit synchronisiert. BS2000/OSD ist ohne Systemzeit nicht ablauffähig.

Die Systembetreuung definiert mit den Parametern für Zeitzone, Zeitsprung, Sommer- bzw. Winterzeit sowie mit der Festlegung von Umstellungsdaten die Basis zur Berechnung der universellen Weltzeit UTC aus der lokalen Zeit LT und zur Sommer-/Winterzeitumstellung der lokalen Zeit während des Systemlaufs.
Damit können System und Benutzer mit der Funktion GTIME sowohl auf ein lokales als auch auf ein systemübergreifend verständliches Zeitbezugssystem zugreifen.

### Automatische Zeitbestimmung bei Startup

Die Zeitbestimmung erfolgt automatisch, wenn folgende Bedingungen erfüllt sind:

- Die IPL-Option UNLOCK ist nicht gesetzt, d.h. die Systemeinleitung erfolgt im Modus FAST oder AUTOMATIC bzw. im Modus DIALOG ohne die IPL-Option UNLOCK.
- Die Zeit der SVP-Uhr ist gültig.
- Die Zeit der SVP-Uhr ist formal richtig (z.B. nicht 31.2.95).
- Die Zeit der SVP-Uhr ist größer als die Zeit des letzten Systemlaufs und der positive Zeitsprung gegenüber dem letzten Systemlauf beträgt nicht mehr als 6 Tage (Ausnahme: im Modus AUTOMATIC kann der positive Zeitsprung beliebig sein).

Erfolgt ein Shutdown unmittelbar vor und der anschließende Startup unmittelbar nach der Umstellung von Sommerzeit auf Winterzeit, beginnt die Systemeinleitung mit einer kleineren Zeit als die des letzten Systemlaufs. Dies ist nur möglich, wenn im Modus DIALOG die Option UNLOCK gesetzt wird (siehe Seite 47).

Negative Zeitsprünge dürfen außerhalb dieser Umstellungszeiten nur in Absprache mit der Systembetreuung durchgeführt werden. Die Konsistenz der Datenhaltung kann damit gefährdet werden.

### Beeinflussbarkeit der Zeitbestimmung bei DIALOG-Startup

Im Modus DIALOG und bei gesetzter Option UNLOCK wird der Operator auch bei einem gültigen Wert der SVP-Uhr zur Bestätigung aufgefordert. Wird die Zeit (und das Datum) der SVP-Uhr nicht bestätigt, muss der Operator selbst Datum und Uhrzeit angeben. Dabei können auch große positive und negative Zeitsprünge gegenüber dem letzten Systemlauf angegeben werden.
Diese so angegebene Zeit wird, wenn sie nach nochmaliger Protokollierung bestätigt wurde, als gültige Systemzeit für den einzuleitenden Systemlauf verwendet.

### 2.1.2 Format der Meldungen bei der Systemeinleitung

Jede Meldung der Systemeinleitung ist in einer der folgenden Formen aufgebaut:

```
? P.msgtext
? tsn-makz.hhmmss % msgtext
```

Dabei ist:

`?` Meldung mit Antwort-Anforderung (ohne Antwort-Anforderung: `%`)

`P` Meldung wird vom Startup ausgegeben

`msgtext`
Meldungstext

`tsn` Meldung wird von BS2000/OSD ausgegeben; tsn ist die Auftragsnummer einer Systemtask oder ein Auftragsname

`-` Trennstrich; entfällt bei Meldungen des residenten Teils der Systemeinleitung

`makz` Meldungsauftragskennzeichen (3 Ziffern oder Buchstaben, führende Nullen dürfen entfallen); makz entfällt bei Meldungen des residenten Teils der Systemeinleitung

`. / #` Trennzeichen; abhängig vom Systemparameter SECSTART (siehe Seite 757)

`hhmmss` Tageszeit; entfällt bei Meldungen des residenten Teils der Systemeinleitung

Der Meldungstext beginnt immer mit einer Schlüsselnummer (3-stellige Meldungsklasse und 4-stellige laufende Nummer, für `x` steht jeweils eine Sedezimalziffer).

Die Meldungen des residenten Teils der Systemeinleitung haben die Schlüsselnummern `NSIxxxx` (Nucleus System Initialization).

Die Meldungen des seitenwechselbaren Teils der Systemeinleitung haben die Schlüsselnummern `EXCxxxx`.

Beispiel einer Meldung des residenten Teils der Systemeinleitung:
`? P.NSI0050  SPECIFY PARAMFILE ...`
Beispiel einer Meldung nach dem residenten Teil der Systemeinleitung:
`? TSC-000.112133%  NSI0077  ENTER AUTOMATIC COMMAND FILE`

Die Antwort selbst beginnt mit der auf das Fragezeichen folgenden Zeichenkette der Meldung, also P. oder tsn (einschließlich eines evtl. vorhandenen Trennstriches, des Meldungsauftragskennzeichens und des darauffolgenden Punktes). Unmittelbar anschließend muss der Antworttext folgen.

Beispiel einer Antwort während des residenten Teils der Systemeinleitung:
`P.FN=PARAMFILE.OPR`
Beispiel einer Antwort nach dem residenten Teil der Systemeinleitung:
`TSC.CMDFILEX`

Bei der Systemeinleitung werden Meldungen stets in englischer Sprache ausgegeben.

### 2.1.3 Hinweise zur Systemeinleitung

Unabhängig vom jeweiligen Server-Typ, der vom Operator zu bedienen ist, sind vorab folgende Fragen zu klären, die zum Verständnis des (korrekten) Ablaufs von Bedeutung sind:

- Wie werden die benötigten Dateien und Geräte identifiziert?
- Was gilt es bezüglich der Eindeutigkeit der Plattenkonfiguration zu beachten?
- Wie wird der Home-Pubset bestimmt und auf Vollständigkeit und Konsistenz geprüft?
- Wie verhält sich das System bei belegten Pubsets oder weggeschalteten Geräten?
- Wodurch können Geräte bei der Systemeinleitung nicht verfügbar sein?
- Was gilt es bei Multiprozessoren zu beachten?
- Welche Rolle spielt die Parameterdatei bei der Systemeinleitung?
- Auf welche Weise wird die Ladeprozedur protokolliert?
- Was ist in Bezug auf die Unicode-Fähigkeit zu beachten?

#### Identifikation der benötigten Dateien und Geräte

Die für HW-IPL bereitzustellenden Datenträger sind wie folgt im jeweiligen SVP-Bildschirm zu adressieren:

Bei S-Servern wird das IPL-Gerät mit der Device Number bezeichnet. Zwischen Device Number und MN existiert eine eindeutige Zuordnung, welche im Handbuch „Systeminstallation“ [57] beschrieben ist.

Bei SQ-Servern wird das IPL-Gerät mit dem mnemotechnischen Gerätenamen (MN) bezeichnet.

Es sollten nur Platten vom Home-Pubset als IPL-Platten verwendet werden. Platten mit einer physikalischen Blockgröße von 4KB werden nicht als IPL-Platte unterstützt.

Alle später benötigten Geräte bei der Ladeprozedur werden mit dem mnemotechnischen Gerätenamen MN angegeben.

Wenn während der Systemeinleitung Dateien verwendet werden, die auf Privatplatten gespeichert sind, so muss neben dem Dateinamen auch die zugehörige VSN angegeben werden. Privatplatten können nur verwendet werden, wenn sie in die Plattenkonfiguration der Systemeinleitung aufgenommen werden.

### Eindeutigkeit der Plattenkonfiguration

Nach SYSBOOT ermittelt SYSIPL die verfügbare Plattenkonfiguration (mittels Online-Scan). Zu diesem Zeitpunkt sind die BS2000-Gerätetabellen, d.h. die generierte HW-Konfiguration, noch nicht verfügbar.

Eine mehrfache Ausführung des Online-Scans wird vermieden, indem für alle identifizierten Platten, die nicht in die Gerätetabelle der Systemeinleitung aufgenommen werden, eine gesonderte Tabelle aufgebaut wird. Im Fall einer späteren Nutzung einer dieser Platten, werden sie in die Gerätetabelle der Systemeinleitung aufgenommen, ohne dass ein erneuter Online-Scan durchgeführt wird.

Privatplatten werden in die Plattenkonfiguration der Systemeinleitung nur dann aufgenommen, wenn die IPL-Platte selbst eine Privatplatte ist, wenn für einen Speicherabzug (SLED) Privatplatten benötigt werden oder wenn der Operator dies explizit anfordert (im Systemeinleitungs-Modus DIALOG mit Auswahl der Option ALLDISK).

Die Systemeinleitung verläuft nur dann problemlos, wenn am vorliegenden Server alle erreichbaren Platten mit einer eindeutigen VSN versehen sind. Wenn mehrere Platten mit derselben VSN erreichbar sind, dann wird dies normalerweise erkannt.
Ausnahme: Das aktuelle IOCF des S-Servers ist nicht korrekt.

Werden Platten mit identischer VSN entdeckt, von denen mindestens eine keine DRV-Platte ist, wird folgendermaßen vorgegangen: Wenn die Platten zu dem selben Pubset gehören wie die IPL-Platte, wird die Platte mit dem selben Zeitstempel wie die IPL-Platte ausgewählt. Besitzt keine der Platten den selben Zeitstempel wie die IPL-Platte oder gehören die Platten zu einem anderen Pubset als die IPL-Platte, dann wird geprüft, ob für eine der Platten bereits einmal eine Auswahl durch die Beantwortung der Meldung `NSI2208` getroffen wurde. Wenn ja, wird wieder diese Platte ausgewählt. Dies gilt sowohl für Pubset-Platten als auch für Privatplatten.



**ACHTUNG!**
Beim oben genannten Ausnahmefall wird nicht erkannt, wenn die reale und generierte Konfiguration nicht übereinstimmen. Bei der Fortsetzung der Systemeinleitung kann es dann zu unvorhersehbaren Ein-/Ausgabe-Fehlern und zur Zerstörung von Datenträgerinhalten kommen!

Bei der Systemeinleitung werden zur Ermittlung der Plattenkonfiguration normalerweise alle angeschlossenen Geräte einer Prüfung unterzogen. Um – gerade bei umfangreicher Peripherie – dieses sehr zeitaufwändige und fehlerträchtige Verfahren (z.B. Ansprechen nicht benötigter und daher nicht betriebsbereiter Geräte) zu umgehen, wird die Möglichkeit geboten, die für die Systemeinleitung tatsächlich benötigte Plattenkonfiguration (d.h. alle Platten des Home- und der Paging-Pubsets und evtl. verwendete Privatplatten) in einer für diesen Zweck auf der IPL-Platte reservierten Datei SYSDAT.IPL-CONF.DSKnnn abzuspeichern (nnn = Nummer der IPL-Platte im Pubset).

*Dynamische Partitionierung der IPL-CONF-Datei*

Die Datei IPL-CONF (SYSDAT.IPL-CONF.DSKnnn) wird dynamisch partitioniert. Dadurch können Startup-Konfigurationen von verschiedenen Servern in dieser Datei gespeichert werden.

Wenn ein Pubset an verschiedenen Servern als Home-Pubset verwendet wird, gehen die Einträge der Startup-Konfiguration in der Datei IPL-CONF nicht verloren, sondern werden jeweils in einer eigenen Partition abgelegt. Beim Wechsel des Servers wird dann die jeweils gültige server-spezifische Startup-Konfiguration verwendet, ohne dass zwingend ein Online-Scan erfolgen muss.

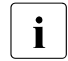
Eine neue Startup-Konfiguration überschreibt eine bestehende Startup-Konfiguration nicht, sondern wird nach der letzten bestehenden als neue eingetragen. Beim Ändern einer bestehenden Startup-Konfiguration für einen bestimmten Server wird die ursprüngliche „verworfen“ und die geänderte Startup-Konfiguration wie eine neue nach der letzten bestehenden eingetragen.

Die Datei IPL-CONF wird jedoch nicht vergrößert: Bei Platzmangel werden sukzessiv zuerst die als ungültig erklärten Startup-Konfigurationen eliminiert. Ist dann der freie Platz in der Datei noch immer nicht ausreichend groß, wird die jeweils am längsten ungenutzte Startup-Konfiguration entfernt. Diese Aktion wird ohne Rückfrage beim Operator durchgeführt und so oft wiederholt, bis der Platz für die neue bzw. geänderte Startup-Konfiguration ausreicht.



**ACHTUNG!**
Das Format der Daten in der IPL-CONF ist mit dem bis BS2000/OSD V6.0 gültigen Datenformat nicht kompatibel. Eine eventuell auf der IPL-Platte bestehende Datei IPL-CONF von BS2000/OSD < V7.0 kann deshalb nicht genutzt werden.
Mit der ersten Systemeinleitung in BS2000/OSD V9.0 von dieser IPL-Platte wird deshalb der Online-Scan ausgeführt und die IPL-CONF neu geschrieben.

Bei einem eventuellen Rückstieg (d.h. die IPL-Platte wird für eine Version < V7.0 IPL-fähig gemacht) und einer erneuten Systemeinleitung wird wiederum ein Online-Scan ausgeführt und die IPL-CONF im bis einschließlich BS2000/OSD V6.0 verwendeten Format neu geschrieben.



In der Datei IPL-CONF können z.B. 12 Startup-Konfigurationen mit jeweils 255 Platten aufgenommen werden. Bei weniger Platten steigt die Anzahl der Konfigurationen: So sind z.B. 30 Konfigurationen mit jeweils 100 Platten oder 57 Konfigurationen mit jeweils 50 Platten möglich.

Das Speichern einer Startup-Konfiguration in der Datei IPL-CONF wird folgendermaßen veranlasst:

- im Modus DIALOG:
  Beantwortet der Operator die Meldung `NSI1110` mit der IPL-CONF-Option SAVE, wird die Startup-Konfiguration als neue Partition gespeichert.
- in allen Startup-Modi:
  Die Startup-Konfiguration wird gespeichert, wenn:
  - die IPL-Platte eine der Platten des Home Pubsets ist
  - und nicht explizit die IPL-CONF-Option IGNORE angegeben wurde
  - und der Online-Scan ausgeführt wurde.

Der Online-Scan wird automatisch immer dann ausgeführt, wenn der Home-Pubset oder einer oder mehrere Paging-Pubsets vergrößert wurde(n) oder ein Paging-Pubset über BS2000-Parameter (in der Datei SYSPAR.BS2.nnn oder über Konsole) neu angegeben wurde.

Die Datei wird dann bei allen folgenden Systemeinleitungen solange automatisch genutzt, wie die Plattenidentifikation VSN der gespeicherten Pubsets, die Geräteidentifikation MN, der Plattentyp und die Menge der zu den Pubsets gehörigen Platten unverändert bleiben.

Um Änderungen an bestehenden, server-spezifischen Startup-Konfigurationen in der Datei SYSDAT.IPL-CONF.DSKnnn zu unterstützen, (z.B. wegen der Verwendung eines vergrößerten oder eines anderen Pubsets), werden – im Modus DIALOG mit der Meldung `NSI1110` – dem Operator bei Angabe der Option IPL-CONF folgende Funktionen angeboten:

- Funktion IGNORE:
  Der Inhalt der Datei SYSDAT.IPL-CONF.DSKnnn wird ignoriert. Ein Online-Scan wird ausgeführt.
- Funktion RESET:
  Die für den Server aktuell gültige Startup-Konfiguration in der Datei SYSDAT.IPL-CONF.DSKnnn wird für die aktuelle Systemeinleitung noch genutzt und anschließend für ungültig erklärt.
- Funktion SAVE:
  Die für Startup benötigte Plattenkonfiguration wird in der Datei SYSDAT.IPL-CONF.DSKnnn als neue Partition abgespeichert.

Alle Kombinationen außer SAVE und RESET sind zulässig.

### Bestimmung des Home-Pubsets und Prüfung auf Vollständigkeit und Konsistenz

Wird bei der Systemeinleitung die Funktion IPL-CONF nicht genutzt, kann – nach der Ermittlung der verfügbaren Plattenkonfigurationen – der Home-Pubset abhängig von der Art der Systemeinleitung wie folgt bestimmt werden:

- im Modus FAST oder AUTOMATIC:
  Es wird der Pubset zum Home-Pubset gewählt, zu welchem die IPL-Platte gehört.
- im Modus DIALOG:
  Falls die IPL-Platte zu einem Pubset gehört, wird der Pubset zum Home-Pubset gewählt, zu welchem die IPL-Platte gehört.
  Falls die IPL-Platte eine Privatplatte ist oder die IPL-Option ALLDISK angegeben wurde und die Plattenkonfiguration aus mehr als einem Pubset besteht, wird die Kennung (PVSID) des Home-Pubsets mit der Meldung `NSI1135` abgefragt.

Platten mit einer physikalischen Blockgröße von 4KB können nicht Platten des Home-Pubsets sein. Auch SM-Pubsets und Shared-Pubsets werden nicht als Home-Pubset unterstützt.

Der Home-Pubset soll hochverfügbar sein, d.h. durch die Hardware wird die Aufzeichnung der Daten für eine logische Platte auf mehreren physikalischen Platten durchgeführt. Durch diese Redundanz tritt beim Ausfall einer physikalischen Platte kein Datenverlust auf.

Alle in der Startup-Konfiguration vorhandenen und nach VSN-Syntax zum Home-Pubset gehörenden Platten werden mit der auf der Pubres vorhandenen Volumeliste verglichen. Im Modus DIALOG oder in Fehlerfällen werden die zum Home-Pubset gehörenden Platten mit dem letzten Zeitstempel und ihrem mnemotechnischen Gerätenamen MN protokolliert (mit der Meldung `NSI1143` oder `NSI1145`).

Folgende Fehlerfälle werden angezeigt:

- Systemkennung (Sysid) des Home-Pubsets nicht definiert (Meldung `NSI1280`)
  Die Systemeinleitung wird mit dem Standardwert fortgesetzt.
- ungültige Sysid des Home-Pubsets (Meldung `NSI1285`)
  Die Systemeinleitung wird mit dem Standardwert fortgesetzt.
- fehlende Platte im Home-Pubset (Meldung `NSI3215`)
  Die Systemeinleitung kann nicht fortgesetzt werden.
- Zeitstempel einer Platte ungleich dem Zeitstempel der Pubres (Meldung `NSI1148`)
  Ob die Systemeinleitung fortgesetzt werden kann, ist abhängig von der Beantwortung der Meldung `NSI1148`.
- DVS-Attribute einer Platte ungleich der DVS-Attribute des Home-Pubsets (`NSI3220`)
  Die jeweiligen DVS-Attribute werden durch die Meldung `NSI3221` angezeigt.
  Die Systemeinleitung wird abgebrochen, da ein derart inkonsistenter Pubset nicht imcatiert werden darf.

- unbekannte Platte im Home-Pubset (Meldung `NSI1145`)
  Nicht zum Home-Pubset gehörende Platten werden mit dem Zusatz `IS NEW` in der Meldung `NSI1145` angezeigt.

**Unterstützung von DRV-Platten**

Auch DRV-Pubsets werden als Home-Pubsets zugelassen. DRV-Privatplatten werden, ebenso wie DRV-Platten außerhalb des Home-Pubsets bei der Startup-Paging-Initialisierung, weiterhin nicht unterstützt.

Da die Funktion DRV in der frühen Phase der Systemeinleitung noch nicht zur Verfügung steht, müssen die Zugriffe auf die DRV-Platten vom IPL-Exec selbst realisiert werden.

Dabei gelten folgende Einschränkungen während der Systemeinleitung:

1. Es wird immer nur von einer Platte eines DRV-Plattenpaares gelesen bzw. geschrieben. Die Auswahl der verwendeten Platte erfolgt entweder automatisch oder im DIALOG-Startup durch den Operator durch Angabe der Option DRV-SELECT.

   Es kann von einem Plattenpaar mit gleicher VSN nur dann eine beliebige Platte ausgewählt werden, wenn beide Platten im Zustand DRV-DUAL sind und der Zeitstempel auf beiden Platten gleich ist. Ist dies nicht der Fall, unterliegt die Auswahl der Platte einigen Einschränkungen, siehe auch Handbuch „DRV“ [17].

   Gehört die IPL-Platte zu einem DRV-Paar, kann die Systemeinleitung nach dem Online-Scan bzw. der Auswertung der IPL-CONF-Datei mit der anderen Platte fortgesetzt werden. Damit werden die Änderungen von Daten nicht mehr auf der beim Start des IPL ausgewählten Platte vorgenommen, sondern auf der anderen Platte des DRV-Paares.

2. Ein-/Ausgabe-Fehler auf eine DRV-Platte bewirken während der Systemeinleitung keinen Rückgriff auf die zweite Platte des DRV-Plattenpaares. Es wird, wie bei Einzelplatten, auf nicht-behebbare Ein-/Ausgabe-Fehler mit Abbruch reagiert.

3. Der Home-Pubset wird im Zustand DRV-MONO hochgefahren; die Einleitung der Rekonstruktion muss mit dem Kommando START-DRV erfolgen.

4. War das Home-Pubset bei Shutdown im Zustand DRV-DUAL, erfolgt die Einleitung der Rekonstruktion durch DRV. Dabei werden nur die seit der Systemeinleitung geänderten Dateien rekonstruiert.

### START/STOP-Modus bei Multiprozessoren

Bei Multiprozessor-Systemen steuert der START/STOP-Modus die Auswahl der Verarbeitungsprozessoren bei Nutzung der SVP-START/STOP-Funktion. Im BS2000-Betriebsablauf wird die SVP-START/STOP-Funktion nicht benötigt. Trotzdem muss der START/STOP-Modus korrekt eingestellt werden, damit bei Nutzung der SVP-START/STOP-Funktion oder der Adress-Stop-Funktion keine unerwünschten Nebeneffekte eintreten.

Für die Einstellung des START/STOP-Modus ist Folgendes gültig:

- Wenn alle Verarbeitungsprozessoren betrieben werden, muss der START/STOP-Modus auf ALL CPU (alle Verarbeitungsprozessoren) gesetzt werden, damit bei STOP alle Verarbeitungsprozessoren angehalten und bei START wieder alle gestartet werden. Dieser Wert ist grundsätzlich als Voreinstellung zu verwenden.
- Wenn nur ein Verarbeitungsprozessor betrieben wird oder wenn bei der Systemeinleitung oder Dump-Funktion die SVP-START/STOP-Funktion bzw. der Adress-Stop benötigt wird, muss der START/STOP-Modus auf TARGET CPU (laufende CPU) gesetzt werden. Damit wird verhindert, dass mit der SVP-START-Funktion Verarbeitungsprozessoren gestartet werden, welche nicht mehr oder noch nicht von der Software kontrolliert werden.
- Wenn mehr als ein Verarbeitungsprozessor, aber nicht alle Verarbeitungsprozessoren betrieben werden, ist die korrekte Funktion der SVP-START/STOP-Funktion nicht gewährleistet.

### Freigabe belegter Pubsets

Falls die zur Systemeinleitung benötigten Home- und/oder Paging-Pubsets beim letzten Systemlauf (mittels Shutdown) nicht ordnungsgemäß exportiert wurden oder falls sie von einem anderen System belegt sind, wird der Operator aufgefordert, diese freizugeben (Meldung `NSI424A` ).
In einer dieser Aufforderung vorausgehenden Meldung wird die Art der Belegung dem Operator genau angezeigt.

Die eindeutige Vergabe von Sysids für mehrere parallel betriebene Home-Pubsets ist zwingend erforderlich.

Die Freigabe darf nur dann erfolgen, wenn diese Pubsets zu diesem Zeitpunkt nicht von einem anderen System verwendet werden. Bei Freigabe ohne sorgfältige Prüfung können die Daten der Pubsets durch uneingeschränkten Zugriff zweier Systeme zerstört werden.

#### Parameterservice

Zur korrekten Durchführung der Systemeinleitung wird die Verwendung einer Startup-Parameterdatei empfohlen. Mit dem Parameterservice werden die Software-Komponenten mit Daten versorgt.

Zur genauen Beschreibung des Parameterservice mit Aufbau und Inhalt der Parameterdateien siehe Kapitel „Parameterservice“ auf Seite 75.

#### Protokollierung des Startup-Dialogs an der IPL-Bedienstation

Alle Meldungen, auch solche, die in den Modi FAST und AUTOMATIC nicht an der Bedienstation ausgegeben werden, sowie alle Antworten werden im Hauptspeicher protokolliert und ab Verfügbarkeit der Funktion CONSLOG in die Protokolldatei übernommen: $SYSAUDIT.SYS.CONSLOG.datum.session-nr.lfd-nr.

Zu Aufbau und Inhalt der Protokolldatei CONSLOG siehe das „Diagnosehandbuch“ [14].

Falls die Systemeinleitung vor Verfügbarkeit der Funktion CONSLOG beendet wird, ist der Bedienstationsdialog in einem anschließenden Dump vorhanden.

### 2.1.4 Systemeinleitung an SQ-Servern

Auf SQ-Servern (X86-64-Architektur) wird die Konfiguration der Bus- und FC-Peripherie über das Trägersystem X2000 festgelegt und beim Startup dynamisch ermittelt.
SQ-Server haben keine Kanalperipherie, eine Hardware-Generierung mit IOGEN entfällt.

| Dateiname | Nutzung |
|---|---|
| statt SYSPRG.IPL.180:<br>SKMPRG.IPL.180 | Ladeobjekte SYSBOOT, SYSIPL + SYSIPLEX und SLED, Standarddatei |
| statt SYSPRG.STRT.180:<br>SKMPRG.STRT.180 | Ladeobjekt SYSSTART + SYSIPLEX |
| statt SYSPRG.BS2.180:<br>SKMPRG.BS2.180 | Ladeobjekt BS2000-CL1/2-Exec |

Tabelle 1: Abweichende Standarddateinamen an SQ-Servern

#### Automatischer IPL und zeitgesteuerter Startup/Shutdown

Auf SQ-Servern kann die Systembetreuung (über den SQ-Manager) in X2000 einen automatischen IPL, einen zeitgesteuerten Startup oder Shutdown konfigurieren. Der Systemparameter SHUTPROC (siehe Seite 758) bestimmt bei einer Systembeendigung über X2000, ob der Shutdown sofort oder durch einen Enter-Job ausgeführt wird.
Falls der automatische IPL eingestellt wurde, wird nach Hochfahren des X2000 automatisch ein IPL von der voreingestellten IPL-Platte eingeleitet.

## 2.2 Arten der Systemeinleitung

Der Operator kann durch eine Vorgabe im SVP-Menü bzw. bei SQ-Servern über den BS2000-Konsolbildschirm unter X2000 die Art der Systemeinleitung einstellen und damit wählen, ob sie komfortabel oder flexibel ablaufen soll. Für einen komfortablen Ablauf stehen die Modi FAST und AUTOMATIC zur Verfügung, für einen flexiblen Ablauf der Modus DIALOG.

Die Sicherstellungsdatei (Save-Datei) hat den versionsunabhängigen Namen SYS.NSI.SAVEREP. Sie wird – außer bei Privatplatten – beim Einrichten einer IPL-Platte von SIR angelegt.

Zu den Dateinamen für die Parameterdatei siehe Kapitel „Parameterservice“ auf Seite 75.

### 2.2.1 FAST-Startup

Im Modus FAST läuft die Systemeinleitung dialogfrei ab. Dafür müssen die zu ladenden Programme, Objektcodekorrekturen (Reps) und Parameter in Dateien mit versionsspezifischen Standardnamen gespeichert sein oder die Namen der Dateien in der Parameterdatei hinterlegt werden. Die Parameterdatei selbst muss mit dem versionsspezifischen Dateinamen mit oder ohne Suffix bezeichnet sein.

Auf der IPL-Konsole werden die wichtigsten Ereignisse der Systemeinleitung protokolliert. Das ausführliche Protokoll – wie im Modus DIALOG – wird in die Protokolldatei CONSLOG geschrieben.

Fehlen die Dateien SYSREP.BS2.180, SYSREP.STRT.180, SYSPAR.BS2.180[.name], SYSPRG.STRT.180 bzw. SYSPRG.BS2.180 und ist in der Parameterdatei keine Rep-Datei oder kein BS2000/OSD angegeben, dann können die Korrekturdaten und Parameter wahlweise von Plattendateien mit beliebigen Namen oder über die Bedienstation im Operatordialog nachgereicht werden (siehe Kapitel „Parameterservice“ auf Seite 75 und Abschnitt „Systemkorrekturen“ auf Seite 60).

Die Wahl des Startup-Typs und die Entscheidung über eine evtl. Rekonstruktion des Benutzerkatalogs ist von den vorher eingestellten Werten der Systemparameter RECONUC und STUPTYPE abhängig (siehe Seite 756 und Seite 761).
Zur Wahl des Startup-Typs siehe auch Abschnitt „Die Auswahl von Startup-Typ und Katalog-Rekonfiguration“ auf Seite 52.

Folgende Standarddateinamen sind bei dialogfreier Systemeinleitung gültig und müssen unter der Benutzerkennung $TSOS auf dem Home-Pubset katalogisiert sein:

| **Dateiname** | **Nutzung** |
|---|---|
| SYSPRG.BOOT.DSKnnn.SAVE | Sicherstellungsbereich [1] |
| SYSPRG.IPL.DSKnnn | Ladeobjekte SYSBOOT, SYSIPL und SLED, beim Startup verwendet [2] |
| SYSREP.IPL.DSKnnn | Reps für SYSIPL, beim Startup verwendet [2] |
| SYSDAT.IPL-CONF.DSKnnn | Sicherstellungsbereich für die Funktion IPL-CONF [1] |
| SYSPRG.STRT.180 | Ladeobjekt SYSSTART |
| SYSREP.STRT.180 | Reps für SYSSTART |
| SYSREP.BS2.180 | Objektcode-Korrekturen (Reps) [3] |
| SYS.NSI.SAVEREP | Zwischenspeicher für Rep-Protokollierung [4] |
| SYSPAR.BS2.180[.name] | Parameter |
| SYSPRG.BS2.180 | Ladeobjekt BS2000-CL1/2-Exec [3] |

[1] Die Datei wird von SIR als leere Sicherungsdatei angelegt und im SVL verankert. „nnn“ im Namensbestandteil „DSKnnn“ entspricht der Nummer der Platte, auf der die Datei von SIR angelegt wurde. Diese Dateinamen werden beim Startup nicht geprüft. Dateien mit einem platten-spezifischen Namen werden bei einer logischen Sicherung nicht berücksichtigt.

[2] Die Datei wurde durch SIR von einer versionsabhängigen Lieferdatei in diese versionsunabhängige Datei kopiert (Suffix „DSKnnn“) und im SVL verankert. „nnn“ im Namensbestandteil „DSKnnn“ entspricht der Nummer der Platte, auf der die Datei von SIR angelegt wurde. Diese Dateinamen werden beim Startup nicht geprüft. Dateien mit einem platten-spezifischen Namen werden bei einer logischen Sicherung nicht berücksichtigt.

[3] Der Dateiname kann mittels Startup-Parameterservice vorgegeben werden.

[4] Die Datei wird von SIR als leere Sicherungsdatei angelegt. Fehlt diese Datei oder ist sie zu klein, hat dies Auswirkung auf den „Sicheren Systemstart“.

Falls eine der Dateien nicht vorhanden ist oder nicht genutzt werden kann, wird der Dateiname über Bedienstation angefordert (nicht bei verankerten Dateien).

Durch die platten-spezifische Benennung der verankerten, d.h. im SVL hinterlegten Dateien können auch mehrere IPL-Platten in einem Pubset parallel eingerichtet werden.

### 2.2.2 AUTOMATIC-Startup

Der Modus AUTOMATIC unterstützt den unbedienten Betrieb. In Verbindung mit einem automatischen Hochfahren des Systems nach abnormaler Systembeendigung und eingeschaltetem automatischen Restart wird der AUTOMATIC-Startup angewendet. Seine Eigenschaften und die Protokollierung entsprechen dem Modus FAST.

Der Modus AUTOMATIC darf nur verwendet werden, wenn die SVP-Uhr funktioniert (dieser Punkt entfällt bei Restart) und die Platten des Home-Pubsets und die Paging-Platten nur von diesem System benutzt werden.

Im Modus AUTOMATIC werden in bestimmten Fehlersituationen Standardreaktionen durchgeführt. Vom Operator wird in diesem Fall keine Entscheidung angefordert.
Ein Beispiel für solche Reaktionen ist die automatische Freigabe benötigter Platten, die wegen abnormaler Systembeendigung noch gesperrt sind.

Zur Beschreibung des automatischen Restart siehe Abschnitt „Automatischer Restart“ auf Seite 74.

Die Wahl des Startup-Typs und die Entscheidung über eine evtl. Rekonstruktion des Benutzerkatalogs ist von den vorher eingestellten Werten der Systemparameter RECONUC und STUPTYPE abhängig (siehe Seite 756 und Seite 761).
Zur Wahl des Startup-Typs siehe auch Abschnitt „Die Auswahl von Startup-Typ und Katalog-Rekonfiguration“ auf Seite 52.

### 2.2.3 DIALOG-Startup

Im Modus DIALOG kann der Operator die Systemeinleitung selbst beeinflussen und spezielle Funktionen auswählen. In einem ausführlichen Dialog erhält der Operator Meldungen am Bildschirm, durch die die erlaubten Eingaben angezeigt werden. Mit ihrer Beantwortung wählt der Operator im Rahmen der gegebenen Möglichkeiten die gewünschten Funktionen aus. Diese Funktionen können sein:

- Freie Auswahl der IPL-Platte, der Eingabemedien für die zu ladenden Programme, Objektcodekorrekturen (Reps) und Parameter
- Aktivieren von bestimmten Testfunktionen oder Funktionen zur Behandlung von Ausnahmesituationen

Weil der Operator mit dem DIALOG-Startup die Vorgaben der Systembetreuung für die Systemeinleitung (Parameter, Reps) ändern kann, wird als Sicherheitsfunktion ein spezieller Schutz für den DIALOG-Startup angeboten:
Die Systemeinleitung von BS2000/OSD im Modus DIALOG kann optional auf eine bestimmte Konsole eingeschränkt werden. Diese Einschränkung kann nur im Modus DIALOG eingestellt und wieder aufgehoben werden; sie bezieht sich stets auf die aktuell verwendete Konsole. Als Berechtigungsschlüssel wird der mnemotechnische Gerätename der Konsole im Home-Pubset hinterlegt.

#### Ablauf des DIALOG-Startup

Der Ablauf des DIALOG-Startups wird ausführlich auf der IPL-Konsole protokolliert.

Im Modus DIALOG erhält der Operator die folgende Meldung, die ihn zur Eingabe von Optionen auffordert:

```
NSI1110 ENTER OPTIONS OR EOT.
        REPLY (UNLOCK,TEST,ALLDISK,DRV-SELECT,CREATE-DRV,IPL-CONF)
```

Es sind beliebige Kombinationen möglich. Die Optionen haben im Einzelnen folgende Bedeutung:

UNLOCK: bewirkt, dass in Konfliktsituationen bei der Bestimmung der Systemzeit fortgesetzt werden bzw. die Bestimmung der Systemzeit durch den Operator erfolgen kann.

TEST: bewirkt, dass dem Operator mit der Meldung `NSI1113` die Eingabe von Optionen für bestimmte Testsituationen angeboten wird.

```
NSI1113   ENTER TEST-OPTIONS OR EOT.
REPLY (DUMPTEST, STRTNAME, REPERRIGN, WATCH, IPLTEST, ASYNSKPIN)
```

*Bedeutung der Test-Optionen*

DUMPTEST:
Testoption, dient nicht zur Systemeinleitung.

STRTNAME:
bewirkt, dass vor dem Laden von SYSSTART die Dateinamen für das Ladeobjekt und für die SYSSTART-Rep-Datei erfragt werden.

REPERRIGN:
Testoption, dient nicht zur Systemeinleitung.

IPLTEST:
Testoption, dient nicht zur Systemeinleitung.

WATCH:
Testoption, dient nicht zur Systemeinleitung. Sie unterstützt die Fehlerdiagnose beim Online-Scan. Für ein bestimmtes Gerät wird eine Überwachung aktiviert, die die Systemeinleitung bei Problemen mit diesem Gerät abbricht. Somit können Diagnoseunterlagen (SLED) genau zum Fehlerzeitpunkt erstellt werden.

ASYNSKPIN:
Testoption, dient nicht zur Systemeinleitung.

ALLDISK: In der von Startup genutzten Plattenkonfiguration werden auch Privatplatten erfasst (die Erfassung erfolgt durch einen Online-Scan), so dass Ladeobjekte und Rep- bzw. Parameterdateien auch von Privatplatte gelesen werden können. Wenn die IPL-Platte eine Privatplatte ist, dann wird die Option ALLDISK automatisch gesetzt.

Es wird implizit die IPL-CONF-Option IGNORE gesetzt. Die Verwendung dieser Option ist nur möglich, wenn max. 1290 Platten online sind.
Im Fehlerfall wird folgende Meldung ausgegeben:

```
NSI2335 OVERFLOW OF STARTUP DEV.TABLE; SWITCH UNNEEDED DISKS OFF
REPLY (R(RETRY); S(STOP))
```

Je nach Antwort wird der Online-Scan wiederholt oder die Systemeinleitung abgebrochen.

DRV-SELECT:

bewirkt, dass die Auswahl der für Startup verwendeten Platten eines DRV-Plattenpaares vom Operator getroffen werden soll.

Dies ist sinnvoll, wenn nach einem Plattenfehler auf einer DRV-Platte während des Startups beim neuen Startup auf die andere Platte eines DRV-Plattenpaares ausgewichen werden soll.

Es wird implizit die IPL-CONF-Option IGNORE gesetzt.

CREATE-DRV:

bewirkt, dass durch implizites Starten des Subsystems DRV nach der Systemeinleitung aus einem SRV-Home-Pubset (Single Recording by Volume) ein DRV-Pubset im Zustand DRV-MONO gemacht wird.

IPL-CONF: mit dieser Option wird die Behandlung der server-spezifischen Partition der auf der IPL-Platte immer eingerichteten Datei SYSDAT.IPL-CONF.DSKnnn gesteuert (zur Bedeutung dieses Parameters siehe auch Seite 39). Wenn diese Option nicht angegeben wird, wird wie in allen anderen Startup-Modi (FAST, AUTOMATIC) versucht, die server-spezifische Partition der Datei auszuwerten.

Bei erfolgreicher Abarbeitung der server-spezifischen Partition wird der Online-Scan unterdrückt.

Bei Angabe dieser Option wird der Operator mit der Meldung `NSI1116` aufgefordert, die Bearbeitung der server-spezifischen Partition in der Datei SYSDAT.IPL-CONF.180 festzulegen:

```
NSI1116  ENTER IPL-CONF OPTION OR EOT.
REPLY (IGNORE; RESET; SAVE. EOT=IGNORE)
```

*Bedeutung der Angaben*

IGNORE:
Die server-spezifische Partition in der Datei SYSDAT.IPL-CONF.DSKnnn wird ignoriert. Ein Online-Scan wird ausgeführt. Diese Option wird implizit gesetzt, wenn die Option ALLDISK angegeben wurde.

RESET:
Die für diesen Server aktuell gültige Partition in der Datei SYSDAT.IPL-CONF.DSKnnn wird für die aktuelle Systemeinleitung noch genutzt und anschließend für ungültig erklärt.

SAVE:
Die für Startup benötigte Plattenkonfiguration wird in der Datei SYSDAT.IPL-CONF.DSKnnn als neue Partition abgespeichert.

Alle Kombinationen außer SAVE und RESET sind zulässig.

Nach Verarbeitung der eingegebenen Optionen ermittelt die Startup-Routine die verfügbare Plattenkonfiguration. Zu diesem Zweck wird entweder eine auf der IPL-Platte vorhandene und gültige Partition in der Datei SYSDAT.IPL-CONF.DSKnnn ausgewertet (falls nicht durch die Option IGNORE explizit ausgeschlossen) oder der Online-Scan ausgeführt. Falls dabei Platten dieselbe VSN haben, muss die Plattenkonfiguration auf Platten mit eindeutiger VSN eingeschränkt werden (siehe dazu Abschnitt „Eindeutigkeit der Plattenkonfiguration“ auf Seite 37).

Falls an dem System mehrere Pubsets verfügbar sind, wird der Operator aufgefordert, das Home-Pubset zu bestimmen (NSI1135). Außer Home-Pubset und Prozessoren werden dann noch Datum und Uhrzeit protokolliert. Gegebenenfalls wird der Operator zur Bestätigung oder Korrektur aufgefordert.

Zum Laden von SYSSTART wird stets der Standard-Dateiname herangezogen.
Der Operator erhält nur dann die folgende Aufforderung, den Dateinamen anzugeben, wenn der Standard-Dateiname im Katalog nicht gefunden wird, oder bei der Eingabe der Test-Optionen explizit STRTNAME angegeben wurde:

```
NSI1190 ENTER STARTUP-FILENAME.
        REPLY (FILENAME(,VOL=VSN); EOT (USE STANDARD FILE))
```

Für die SYSSTART-Rep-Datei wird der Standard-Dateiname verwendet.
Ist diese Datei nicht vorhanden oder wurde die Option STRTNAME gewählt, wird der Name der Datei bzw. das Eingabemedium über die Meldung NSI0050 erfragt (siehe Seite 66).

Danach wird der Dateiname und das Eingabemedium für die Parameter erfragt (siehe auch Kapitel „Parameterservice“ auf Seite 75). Anschließend erscheint wieder die Meldung:

```
NSI1190 ENTER BS2000-FILENAME.
        REPLY (FILENAME(,VOL=VSN); EOT (USE STANDARD FILE))
```

Der Operator kann den Dateinamen des zu ladenden BS2000-Systems eingeben und die Archivnummer, falls die Datei sich auf Privatplatte befindet. Wird kein Dateiname angegeben, wird die Parameterdatei ausgewertet. Ist auch dort kein Dateiname eingetragen, wird BS2000/OSD aus der Standarddatei SYSPRG.BS2.180 geladen.

Im Anschluss erfolgt die Repverarbeitung (siehe Seite 66), dann werden die ausgewählten Seitenwechselplatten protokolliert.

Mit folgender Meldung wird der Operator zur Angabe des Namens einer Kommandodatei aufgefordert, deren Kommandos nach „System Ready“ automatisch ausgeführt werden:

```
NSI0077 ENTER AUTOMATIC COMMAND FILE NAME.
        REPLY (FILE NAME; N(DO NOT USE); EOT(USE STANDARD))
```

Mit folgender Meldung wird der Operator aufgefordert, die Art des Systemstarts (siehe Seite 52) anzugeben:
(siehe auch die Beschreibung des Systemparameters STUPTYPE, Seite 761)

```
NSI6005 SYSTEM PARAMETER STUPTYPE = (&00). SHALL VALUE BE CHANGED?
        REPLY ( U(NCHANGED), W(ARM), C(OLD), S(ELECTIVE), Z(IP),
        T(SN FILE RESET ONLY), J(OIN AND TSN FILE RESET), EOT=UNCHANGED)
```

Durch Beantwortung folgender Meldung kann durch den Operator bestimmt werden, ob und wie eine Benutzerkatalog-Rekonstruktion durchgeführt werden soll (siehe dazu die Beschreibung des Systemparameters RECONUC, Seite 756).

```
NSI6010 SYSTEM PARAMETER RECONUC = (&00). SHALL VALUE BE CHANGED?
        REPLY ( U(NCHANGED), N(O), B(ACKUP), T(SOSCAT), A(LL), R(ESET),
        EOT=UNCHANGED )
```

Der weitere Ablauf der Systemeinleitung ist durch DSSM gesteuert (siehe auch Hinweis auf Seite 29). Mit der Meldung „System Ready“ ist die Systemeinleitung beendet.

### 2.2.4 Systemeinleitung mit wechselndem Modus

Um auch in Sonderfällen – Erprobung neuer Software, Parameter oder Korrekturen – die Systemeinleitung möglichst dialogfrei ablaufen zu lassen, kann in der Parameterdatei neben den Abweichungen vom Standardfall auch die Option verankert werden, nach Abarbeitung der Vorgaben in einem anderen Modus fortzufahren. Der Operator wählt also im SVP-Menü den Modus DIALOG und gibt in dessen Verlauf die passende Parameterdatei an. Mit dem Parameterdatei-Eintrag NEW-IPL-MODE=FAST unter dem Schlüsselwort SYSOPT-IPL kann die Systembetreuung dann vorgeben, dass nach Auswertung der Parameterdatei die Systemeinleitung im Modus FAST, also unbedient, fortgesetzt wird.

Zum Parametersatz SYSOPT-IPL siehe auch Abschnitt „Änderung von IPL-Optionen (SYSOPT-IPL)“ auf Seite 123.

### 2.2.5 Die Auswahl von Startup-Typ und Katalog-Rekonfiguration

Die Entscheidung, welcher Systemstart gewählt werden soll, trifft der Operator beim DIALOG-Startup durch Beantworten der Meldung

```
NSI6005 SYSTEM PARAMETER STUPTYPE = (&00). SHALL VALUE BE CHANGED?
        REPLY ( U(NCHANGED), W(ARM), C(OLD), S(ELECTIVE), Z(IP),
        T(SN FILE RESET ONLY), J(OIN AND TSN FILE RESET), EOT=UNCHANGED)
```

Bei den Modi FAST und AUTOMATIC kann diese Entscheidung nur durch Belegung des Systemparameters STUPTYPE getroffen werden (siehe Seite 761).

Beim DIALOG-Startup sind folgende Antworten möglich:

- U=Unchanged

  Der Systemstart erfolgt wie von der Systembetreuung durch den Systemparameter STUPTYPE vereinbart.

- W=Warmstart

  Der Warmstart ist die Standardvorbelegung des Systemparameters STUPTYPE.

  Alle Jobs, die noch in den JMS-Auftragswarteschlangen sind (Verzeichnis mit Benutzerkennungen und Auftragsfolgenummern für Type 1 und Type 2) und nicht während des vorangegangenen Systemlaufs bearbeitet wurden, werden jetzt durch die Auftragssteuerung wieder zur Verarbeitung bereitgestellt.

- C=Kaltstart

  Bei diesem Typ des Systemstarts bleiben Jobs in der Warteschlange unberücksichtigt. Sie werden ohne Bearbeitung gelöscht. Wenn z.B. ein Plattenfehler in den Auftragswarteschlangendateien einen Systemabbruch verursacht, muss der nächste Systemlauf immer mit einem Kaltstart eingeleitet werden. Denn nach einem Plattenfehler muss die Warteschlange vollständig neu aufgebaut werden. Der Kaltstart sollte nach Möglichkeit vermieden werden, da dadurch Termin- und Repeat-Jobs verloren gehen. Für Termin- und Repeat-Jobs können vor dem Kaltstart mit dem Dienstprogramm JMP die erforderlichen ENTER-JOB-Kommandos rekonstruiert werden (siehe Abschnitt „JMP: Rekonstruktion von Batch-Jobs“ auf Seite 471).

- T=First-Start mit Rücksetzen der TSN-Datei

  Dieser Typ des Systemstarts muss bei der erstmaligen Benutzung eines Pubsets verwendet werden.

  Das Zurücksetzen der TSN-Datei ($TSOS.TSOSJOIN.TSNN) bewirkt, dass die JMS-Auftragswarteschlange und die TSN-Datei zurückgesetzt werden.
  D.h. existierende Jobs werden gelöscht, die Session-Nummer wird auf 1 zurückgesetzt und die nächste Auftragsnummer (tsn), die das System vergibt, ist – je nach TSN-Modus – 0001 oder 0AAA.

- J=First-Start mit Rücksetzen des Benutzerkatalogs

  Es werden der Benutzerkatalog, die JMS-Auftragswarteschlange und die TSN-Datei zurückgesetzt.

  First-Start ist nur notwendig beim ersten Hochfahren nach einer Erstinstallation (vom Starterband oder – für SQ-Server – von der „Setup-CD“).

  Das Zurücksetzen des Benutzerkatalogs bewirkt, dass ein vorhandener alter Benutzerkatalog gelöscht und ein neuer erzeugt wird. Dieser neue Benutzerkatalog enthält nur Einträge für die Systemkennungen (SERVICE, SYSAUDIT, SYSDUMP, SYSGEN, SYSHSMS, SYSMAREN, SYSNAC, SYSPRIV, SYSSNAP, SYSSNS, SYSSOPT, SYSSPOOL, SYSUSER, SYSROOT, SYSOPR und TSOS).

  **ACHTUNG!**
  Alle Dateien außer denen auf der Kennung TSOS werden gelöscht!
  Außer zu den Systemdateien ist kein Zugriff (über den Katalog) zu sämtlichen Benutzerdateien mehr möglich. Auch der von diesen gelöschten Benutzerdateien zuvor belegte Speicherplatz wird freigegeben.

  Die Benutzerkennungen des Systems (mit Ausnahme von TSOS und SERVICE) erhalten die Abrechnungsnummer SYSACC und sind gesperrt. Durch das Kommando UNLOCK-USER können diese Kennungen von der Systembetreuung zugänglich gemacht werden. Sie sollten dann jedoch mit Zugangsschutzattributen versehen werden.

  Das Zurücksetzen der TSN-Datei ($TSOS.TSOSJOIN.TSNN) bewirkt, dass die JMS-Auftragswarteschlange und die TSN-Datei zurückgesetzt werden.
  D.h. existierende Jobs werden gelöscht, die Session-Nummer wird auf 1 zurückgesetzt und die nächste Auftragsnummer (tsn), die das System vergibt, ist – je nach TSN-Modus – 0001 oder 0AAA.

- S=Selective-Start

  Bei diesem Typ des Systemstarts kann der Operator bestimmte Warteschlangen angeben, die erhalten bleiben sollen, während die übrigen vom System wie bei einem Kaltstart behandelt werden.

  Nach der Antwort `S` auf die Meldungen `NSI6005` und `NSI6010` erscheint die Meldung

  ```
  ? JOBP... JMS0510 SPECIFY QUEUE(S) OF BATCH JOBS TO BE SAVED WITH
            SELECTIVE START. REPLY: (1: AWAITING PROCESSING; 2: BEING
            PROCESSED)
  ```

  Durch Beantworten der Meldung `JMS0510` können die JMS-Jobs (in den Warteschlangen Q1 und Q2) angegeben werden, die erhalten bleiben sollen.

Das Format der Antwort ist: `JOBP.n`

wobei
n=1 die Auftragswarteschlange für zu verarbeitende Batch-Jobs und
n=2 die Batch-Jobs, die in Bearbeitung waren, bezeichnet.

Bei Angabe von `JOBP.` werden keine Warteschlangen ausgewählt.

*Beispiel*

```
JMS0510 SPECIFY QUEUE(S) OF BATCH JOBS TO BE SAVED WITH SELECTIVE
        START. REPLY: (1: AWAITING PROCESSING; 2: BEING PROCESSED)
```

```
JOBP.1
```

Danach werden die alten Warteschlangendateien (JOBPOOL-Datei für die JMS-Jobs in den Warteschlangen Q1 und Q2) daraufhin geprüft, ob sie Jobs enthalten, die erhalten bleiben sollen. Ist dies der Fall, so werden die Einträge der Auftragsbearbeitung in die neue Warteschlangendateien übernommen und die zugehörigen Zähler entsprechend gesetzt. Wenn die Dateien gelesen worden sind, werden sie geschlossen und aus dem Katalog gestrichen. Die neuen Dateien werden im neuen Systemlauf abgearbeitet.

- Z=Zip-Start

  Dieser Typ des Systemstarts wird angewendet, wenn für einen normalen Start nicht mehr genug freier Plattenspeicherplatz zur Verfügung steht. Bei der Antwort `Z` auf die Meldungen `NSI6005` und `NSI6010` wird kein Versuch unternommen, Plattenspeicherraum für Warteschlangendateien zuzuweisen. Diese Betriebsart ist nur vorgesehen, um nach dem Einleiten des Systems Dateien auszulagern oder zu löschen, um wieder freien Speicherplatz auf den Platten zu erhalten. In diesem Systemlauf sollten keine anderen Aufgaben durchgeführt werden.

Durch Beantwortung folgender Meldung kann im DIALOG-Startup in analoger Form durch den Operator bestimmt werden, ob und wie eine Rekonstruktion des Benutzerkatalogs durchgeführt werden soll:
(siehe dazu die Beschreibung des Systemparameters RECONUC, Seite 756)

```
NSI6010 SYSTEM PARAMETER RECONUC = (&00). SHALL VALUE BE CHANGED?
        REPLY ( U(NCHANGED), N(O), B(ACKUP), T(SOSCAT), A(LL), R(ESET),
        EOT=UNCHANGED )
```

## 2.2.6 Beispiel für einen DIALOG-Startup

Das folgende Beispiel zeigt den Ablauf eines DIALOG-Startups (in Auszügen) an einem S-Server . Die Darstellung entspricht der Protokollierung in der CONSLOG-Datei; für die Ausgabe auf die Bedienstation können sich Abweichungen bezüglich des Meldungsstroms ergeben.

```
TCLOG    .093801  ***2011-11-10*** 000001 **** UTC+01:00 *************************************************************
(CO) %   P-000.093801 NSI00E3 IPL-REPS READ: 0; EXECUTED: 0
(CO) %   P-000.093801 NSI1100 IPL DEVICE = FCH1.0; IPL PATH = 6C80 (MN=6C80)
(CO) %   P-000.093801 NSI1106 *** BS2000 DIALOGUE STARTUP ***
(CO) ?   P-000.093801 NSI1110 ENTER OPTIONS OR EOT. REPLY (UNLOCK,TEST,ALLDISK,DRV-SELECT,CREATE-DRV,IPL-CONF)
   P R(CO)-000.093801. ----------------------------------------------------------------------------------        (1)
(CO) %   P-000.093801 NSI3135 IPL DISK-SETUP READ FROM IPL-CONF PREPARED 2011-11-07 12:01:12
(CO) %   P-000.093801 NSI1143 006 PUBVOLS OF HOME-PVS FCH1 WITH SYSID 101 ONLINE ------------------------------        (2)
(CO) %   P-000.093801 NSI1145 FCH1.0 2011-11-10 09:33:55 ON 6C80
(CO) %   P-000.093801 NSI1145 FCH1.1 2011-11-10 09:33:55 ON 6C81
(CO) %   P-000.093801 NSI1145 FCH1.2 2011-11-10 09:33:55 ON 6C82
(CO) %   P-000.093801 NSI1145 FCH1.3 2011-11-10 09:33:55 ON 6C83
(CO) %   P-000.093801 NSI1145 FCH1.4 2011-11-10 09:33:55 ON 6C87
(CO) %   P-000.093801 NSI1145 FCH1.5 2011-11-10 09:33:55 ON 6C9D
(CO) ?   P-000.093801 NSI1126 CHECK IF AUTHORIZATION FOR DIALOGUE STARTUP SHALL BE RESTRICTED TO THIS CONSOLE CO.
                      REPLY (Y; N; EOT=N)
   P R(CO)-000.093801.
(CO) %   P-000.093801 NSI1153 STATE OF PROCESSORS ONLINE:
(CO) %   P-000.093801 NSI1155 CPU 00 ONLINE, ATTACHED (IPL CPU)
(CO) %   P-000.093801 NSI1158 CPU 01 ONLINE, DETACHED
(CO) %   P-000.093801 NSI1163 LOCAL DATE = 2011-11-10, TIME = 09:36:23 FROM SVP -------------------------------        (3)
(CO) %   P-000.093801 NSI1180 LOAD ADDRESS OF SYSPRG.STRT.180 = 00767000
(CO) %   P-000.093801 NSI00E6 SYSREP.STRT.180 IS USED AS REP FILENAME -----------------------------------------        (4)
(CO) ?   P-000.093801 NSI0050 SPECIFY PARAM FILE OR DEVICE.
                      REPLY (EOT (USE STANDARD FILE);FN=FILENAME,(VOL=VSN);CONS;END)
   P R(CO)-000.093801. ----------------------------------------------------------------------------------        (5)
(CO) %   P-000.093801 NSI00E6 SYSPAR.BS2.180.VM06S210 IS USED AS PARAM FILENAME
(CO) %   P-000.093801 NSI006B FILE SYSPAR.BS2.180.VM06S210 NOT FOUND
(CO) %   P-000.093801 NSI00BC LAST PARAM FILE/DEVICE IGNORED
(CO) %   P-000.093801 NSI00E6 SYSPAR.BS2.180 IS USED AS PARAM FILENAME
(CO) %   P-000.093801 NSI008A PROCESSING PARAMETER ADD FILE PARAMS.ACCOUNT
(CO) %   P-000.093801 NSI008A PROCESSING PARAMETER ADD FILE PARAMS.DSSM.X
(CO) %   P-000.093801 NSI008A PROCESSING PARAMETER ADD FILE PARAMS.ISAM
(CO) %   P-000.093801 NSI008A PROCESSING PARAMETER ADD FILE PARAMS.GTIME
(CO) %   P-000.093801 NSI008A PROCESSING PARAMETER ADD FILE PARAMS.MEMORY
(CO) %   P-000.093801 NSI008A PROCESSING PARAMETER ADD FILE PARAMS.OPR
(CO) %   P-000.093801 NSI008A PROCESSING PARAMETER ADD FILE PARAMS.SYSOPT-CLASS2
(CO) %   P-000.093801 NSI008A PROCESSING PARAMETER ADD FILE PARAMS.SNAPSHOT
(CO) %   P-000.093801 NSI008A PROCESSING PARAMETER ADD FILE PARAMS.SRPM
(CO) %   P-000.093801 NSI008A PROCESSING PARAMETER ADD FILE PARAMS.VM2000
(CO) ?   P-000.093801 NSI0050 SPECIFY PARAM FILE OR DEVICE.
                      REPLY (EOT (USE STANDARD FILE);FN=FILENAME,(VOL=VSN);CONS;END)
   P R(CO)-000.093801.END -------------------------------------------------------------------------------        (6)
(CO) ?   P-000.093801 NSI1190 ENTER BS2000-FILENAME. REPLY (FILENAME(,VOL=VSN); EOT (USE STANDARD FILE))
   P R(CO)-000.093801. ----------------------------------------------------------------------------------        (7)
(CO) ?   P-000.093801 NSI0050 SPECIFY BS2000 REP FILE OR DEVICE.
                      REPLY (EOT (USE STANDARD FILE);FN=FILENAME,(VOL=VSN);CONS;END)
   P R(CO)-000.093801.-----------------------------------------------------------------------------------        (8)
(CO) %   P-000.093801 NSI00E6 SYSREP.BS2.180 IS USED AS REP FILENAME
(CO) ?   P-000.093801 NSI0050 SPECIFY BS2000 REP FILE OR DEVICE.
                      REPLY (EOT (END); FN=FILENAME,(VOL=VSN);CONS;END)
   P R(CO)-000.093801.END -------------------------------------------------------------------------------        (9)
(CO) %   P-000.093801 NSI0028 AVAILABLE MEMORY = 2.5 GB. CPU = 10020001 22600000 ------------------------------       (10)
(CO) %   P-000.093801 NSI5103 BS2000 : NAME = N10BXS, VERSION = 18.0A00, GEN-TIME = 2011-10-26 13:51:00
(CO) %   P-000.093801 NSI5102 COPYRIGHT (C) FUJITSU TECHNOLOGY SOLUTIONS 2009 ALL RIGHTS RESERVED
(CO) %   P-000.093801 EMM2960 EFFECTIVE SYSSIZE VALUE IS 384 MB
(CO) %   P-000.093801 NSI5150 CPU'S ONLINE: 2  CPU'S INSTALLED: 2
```

```
(CO) %   P-000.093801 ETMGT10 GTIME-VALUES FOR SESSION: ZONE=+01:00, DIFF=1:00
(CO) %   P-000.093801 ETMGT01 TODR-EPOCH FOR SESSION: EPOCH=00
(CO) %   P-000.093801 NSI3130 IPL-CONF SAVED ONTO DISK MN=6C80 WITHOUT ERRORS
(CO) %   P-000.093801 NSI5104 IOCF: NAME = S2100001S2100001 / STANDARD OSDV3-8 / 25.10.11
                      GEN-TIME = 2011-10-25 13:28:03
(CO) %   E-000.093801 EMM2301 PAGING AREA ON VOLUME FCH1.2 SIZE = 00500.00 MB.
(CO) %   E-000.093801 EMM2301 PAGING AREA ON VOLUME FCH1.3 SIZE = 00500.00 MB.
(CO) %   E-000.093801 EMM2301 PAGING AREA ON VOLUME FCH1.4 SIZE = 00500.00 MB.
(CO) %   E-000.093801 EMM2301 PAGING AREA ON VOLUME FCH1.5 SIZE = 00500.00 MB.
(CO) %   E-000.093801 EMM2850 THE SIZE OF THE PAGING-AREA IS 0002000.00 MB; THE RESERVED SIZE OF THE PAGING-AREA
                      IS 0000011.25 MB.
(CO) %   P-000.093801 NSI6102 BS2000 EXECUTIVE: CLASS2 ACTIVE ————————————————————————————————————————————————    (11)
(CO) %   P-000.093801 NSI4110 /INPUT DISK F=SYSPAR.BS2.180 ———————————————————————————————————————————————————    (12)
(CO) %   P-000.093801 NSI4110 /BEGIN SYSOPT-CLASS2
(CO) %   P-000.093801 NSI4110 ENCRYPT=Y
(CO) %   P-000.093801 NSI4110 /EOF
(CO) %   P-000.093801 NSI4110 /ADD PARAMS.ACCOUNT
(CO) %   P-000.093801 NSI4110 /ADD PARAMS.DSSM.X
(CO) %   P-000.093801 NSI4110 /ADD PARAMS.ISAM
(CO) %   P-000.093801 NSI4110 /ADD PARAMS.GTIME
(CO) %   P-000.093801 NSI4110 /ADD PARAMS.MEMORY
(CO) %   P-000.093801 NSI4110 /ADD PARAMS.OPR
(CO) %   P-000.093801 NSI4110 /ADD PARAMS.SYSOPT-CLASS2
(CO) %   P-000.093801 NSI4110 /ADD PARAMS.SNAPSHOT
(CO) %   P-000.093801 NSI4110 /ADD PARAMS.SRPM
(CO) %   P-000.093801 NSI4110 /ADD PARAMS.VM2000
(CO) %   P-000.093801 NSI4110 /INPUT DISK F=PARAMS.ACCOUNT
(CO) %   P-000.093801 NSI4110 /BEGIN ACCOUNT
(CO) %   P-000.093801 NSI4110 /STOP-ACCOUNT
(CO) %   P-000.093801 NSI4110 /EOF
(CO) %   P-000.093801 NSI4110 /INPUT DISK F=PARAMS.DSSM.X
(CO) %   P-000.093801 NSI4110 /BEGIN DSSM
(CO) %   P-000.093801 NSI4110 SSMCAT=SYS.SSD.CAT.X
(CO) %   P-000.093801 NSI4110 LOGGING=OFF
(CO) %   P-000.093801 NSI4110 VERSION=043
(CO) %   P-000.093801 NSI4110 /EOF
(CO) %   P-000.093801 NSI4110 /INPUT DISK F=PARAMS.ISAM
(CO) %   P-000.093801 NSI4110 /BEGIN ISAM
(CO) %   P-000.093801 NSI4110 LCLDFPS=96
(CO) %   P-000.093801 NSI4110 GLBPS=512
(CO) %   P-000.093801 NSI4110 LCLPS=96
(CO) %   P-000.093801 NSI4110 DFPPROC=*ADV
(CO) %   P-000.093801 NSI4110 GLBDFPN=4
(CO) %   P-000.093801 NSI4110 /EOF ISAM
(CO) %   P-000.093801 NSI4110 /INPUT DISK F=PARAMS.GTIME
(CO) %   P-000.093801 NSI4110 /BEGIN GTIME
(CO) %   P-000.093801 NSI4110 ZONE=+01:00
(CO) %   P-000.093801 NSI4110 DIFF=1:00
(CO) %   P-000.093801 NSI4110 SEASON=W
(CO) %   P-000.093801 NSI4110 EPOCH=00
(CO) %   P-000.093801 NSI4110 CHDATE=1980-04-06/02:00
(CO) %   P-000.093801 NSI4110 CHDATE=1980-09-28/03:00
. . .
(CO) %   P-000.093801 NSI4110 CHDATE=2019-03-31/02:00
(CO) %   P-000.093801 NSI4110 CHDATE=2019-10-27/03:00
(CO) %   P-000.093801 NSI4110 /EOF
(CO) %   P-000.093801 NSI4110 /INPUT DISK F=PARAMS.MEMORY
(CO) %   P-000.093801 NSI4110 /BEGIN MEMORY
(CO) %   P-000.093801 NSI4110 ASAMCTRL 15
(CO) %   P-000.093801 NSI4110 SHRSIZE 4,UNIT=1MB
(CO) %   P-000.093801 NSI4110 SYSSIZE 128
(CO) %   P-000.093801 NSI4110 WSCTRL 2
(CO) %   P-000.093801 NSI4110 /EOF
(CO) %   P-000.093801 NSI4110 /INPUT DISK F=PARAMS.OPR
(CO) %   P-000.093801 NSI4110 /BEGIN OPR
(CO) %   P-000.093801 NSI4110 DEFINE CONSOLE=C0
```

```
(CO) %   P-000.093801 NSI4110 DEFINE CONSOLE=C1
(CO) %   P-000.093801 NSI4110 DEFINE CONSOLE=C2
(CO) %   P-000.093801 NSI4110 DEFINE CONSOLE=TS,TELESERVICE=YES
(CO) %   P-000.093801 NSI4110 SET-CODE CODE=*ALL,CONSOLE=*IPL
(CO) %   P-000.093801 NSI4110 SET-CODE CODE=X,CONSOLE=ATSC
(CO) %   P-000.093801 NSI4110 SET-CODE CODE=A,CONSOLE=(BCAM,CON1,CON2,ATSC,QM12,ROB1,ROB2,ROB3,ROB4)
(CO) %   P-000.093801 NSI4110 SET-CODE CODE=C,CONSOLE=(BCAM,DADM,CON1,CON2,ATSC,TES1,TES2,CTV@,QM12)
(CO) %   P-000.093801 NSI4110 SET-CODE CODE=D,CONSOLE=(BCAM,CON1,CON2,ATSC,QM12)
(CO) %   P-000.093801 NSI4110 SET-CODE CODE=E,CONSOLE=(BCAM,CON1,CON2,CON3,CON4,CON5,ATSC,QM12)
(CO) %   P-000.093801 NSI4110 SET-CODE CODE=E,CONSOLE=(BCAM,ROB1,ROB2,ROB3,ROB4,CSCO,MARE)
(CO) %   P-000.093801 NSI4110 SET-CODE CODE=G,CONSOLE=(BCAM,CON1,CON2,ATSC,QM12,ROB1,ROB2,ROB3,ROB4)
(CO) %   P-000.093801 NSI4110 SET-CODE CODE=G,CONSOLE=(BCAM,CSCO)
(CO) %   P-000.093801 NSI4110 SET-CODE CODE=H,CONSOLE=(BCAM,CON1,CON2,ATSC,QM12)
(CO) %   P-000.093801 NSI4110 SET-CODE CODE=I,CONSOLE=(BCAM,CON1,CON2,ATSC,QM12)
(CO) %   P-000.093801 NSI4110 SET-CODE CODE=J,CONSOLE=(BCAM,CON1,CON2,ATSC,QM12)
(CO) %   P-000.093801 NSI4110 SET-CODE CODE=K,CONSOLE=(BCAM,CON1,CON2,ATSC,QM12)
(CO) %   P-000.093801 NSI4110 SET-CODE CODE=N,CONSOLE=(BCAM,CON1,CON2,ATSC,QM12,ROB1,ROB2,ROB3,ROB4)
(CO) %   P-000.093801 NSI4110 SET-CODE CODE=O,CONSOLE=(BCAM,CON1,CON2,ATSC,QM12)
(CO) %   P-000.093801 NSI4110 SET-CODE CODE=P,CONSOLE=(BCAM,CON1,CON2,ATSC,QM12,CSCO)
(CO) %   P-000.093801 NSI4110 SET-CODE CODE=R,CONSOLE=(BCAM,CON1,CON2,ATSC,QM12)
(CO) %   P-000.093801 NSI4110 SET-CODE CODE=S,CONSOLE=(BCAM,CON1,CON2,ATSC,QM12)
(CO) %   P-000.093801 NSI4110 SET-CODE CODE=T,CONSOLE=(BCAM,CON1,CON2,ATSC,QM12,CSCO)
(CO) %   P-000.093801 NSI4110 SET-CODE CODE=T,CONSOLE=(ROB1,ROB2,ROB3,ROB4,MARE)
(CO) %   P-000.093801 NSI4110 SET-CODE CODE=U,CONSOLE=(BCAM,CON1,CON2,ATSC,QM12,ROB1,ROB2,ROB3,ROB4)
(CO) %   P-000.093801 NSI4110 SET-CODE CODE=9,CONSOLE=(BCAM,CON1,CON2,ATSC,QM12,CSCO)
(CO) %   P-000.093801 NSI4110 ADD-CMD-ENTRY C-N=CMD1,A-C=R,A-N=CON1
(CO) %   P-000.093801 NSI4110 ADD-CMD-ENTRY C-N=CMD2,A-C=A,A-N=CON2
(CO) %   P-000.093801 NSI4110 ADD-CMD-ENTRY C-N=CMD3,A-C=U,A-N=CON3
(CO) %   P-000.093801 NSI4110 ADD-CMD-ENTRY C-N=CSC,A-C=E,A-N=CSCO
(CO) %   P-000.093801 NSI4110 /EOF
(CO) %   P-000.093801 NSI4110 /INPUT DISK F=PARAMS.SYSOPT-CLASS2
(CO) %   P-000.093801 NSI4110 /BEGIN SYSOPT-CLASS2
(CO) %   P-000.093801 NSI4110 ASRSW2=1
(CO) %   P-000.093801 NSI4110 BMTNUM=64
(CO) %   P-000.093801 NSI4110 CATBUFR=Y
(CO) %   P-000.093801 NSI4110 CMDFILE=CMDFILE
(CO) %   P-000.093801 NSI4110 DMCMAXP=256
(CO) %   P-000.093801 NSI4110 DMMAXSC=24
(CO) %   P-000.093801 NSI4110 DUMPCTRL=X'04'
(CO) %   P-000.093801 NSI4110 DUMPSREF=X'00'
(CO) %   P-000.093801 NSI4110 EACTETYP=1
(CO) %   P-000.093801 NSI4110 EAMMEM=2730
(CO) %   P-000.093801 NSI4110 EAMMIN=10225
(CO) %   P-000.093801 NSI4110 EAMSEC=3000
(CO) %   P-000.093801 NSI4110 EAMSPVS=X'01'
(CO) %   P-000.093801 NSI4110 ENCRYPT=Y
(CO) %   P-000.093801 NSI4110 FMTYFNLG=4
(CO) %   P-000.093801 NSI4110 FSHARING=1
(CO) %   P-000.093801 NSI4110 FST32GB=1
(CO) %   P-000.093801 NSI4110 JTSTDMEM=32
(CO) %   P-000.093801 NSI4110 L4SPDEF=10000
(CO) %   P-000.093801 NSI4110 MCXSPXCS=Y
(CO) %   P-000.093801 NSI4110 MSGFIL01=SYSMES.BS2CP.180
(CO) %   P-000.093801 NSI4110 MSGFIL02=SYSMES.EKP.01
(CO) %   P-000.093801 NSI4110 MSGNOFL=2
(CO) %   P-000.093801 NSI4110 MSGLPRI=ED
(CO) %   P-000.093801 NSI4110 MSGCENTL=200
(CO) %   P-000.093801 NSI4110 NRTKILL=N
(CO) %   P-000.093801 NSI4110 NBMSGCSD=C'Y'
(CO) %   P-000.093801 NSI4110 PWACTIVE=1000
(CO) %   P-000.093801 NSI4110 PWENTERD=1000
(CO) %   P-000.093801 NSI4110 PWERRORS=64
(CO) %   P-000.093801 NSI4110 SSMCOPT=Y
(CO) %   P-000.093801 NSI4110 STUPTYPE=C
(CO) %   P-000.093801 NSI4110 SYSTHOOK=YES
(CO) %   P-000.093801 NSI4110 SYSVECT=2
```

```
(CO) %   P-000.093801 NSI4110 TASKVECT=2
(CO) %   P-000.093801 NSI4110 TEMPFILE=C'#'
(CO) %   P-000.093801 NSI4110 /EOF
(CO) %   P-000.093801 NSI4110 /INPUT DISK F=PARAMS.SNAPSHOT
(CO) %   P-000.093801 NSI4110 /BEGIN SNAP
(CO) %   P-000.093801 NSI4110 SNAP-ACTIVE-SWITCH=ON
(CO) %   P-000.093801 NSI4110 /EOF
(CO) %   P-000.093801 NSI4110 /INPUT DISK F=PARAMS.SRPM
(CO) %   P-000.093801 NSI4110 /BEGIN SRPM
(CO) %   P-000.093801 NSI4110 /EOF
(CO) %   P-000.093801 NSI4110 /INPUT DISK F=PARAMS.VM2000
(CO) %   P-000.093801 NSI4110 /BEGIN VM2000
(CO) %   P-000.093801 NSI4110 VERSION=V09.5A
(CO) %   P-000.093801 NSI4110 /EOF
(CO) %   E-000.093801 EMM2310 THE SIZE OF MAIN MEMORY IS 00002560 MB; THE MINIMAL SIZE OF MAIN MEMORY IS 00002560 MB
(CA) % UCO-000.093803  %  NBR0706 TEST OUTPUT FOR CONSOLE 'CA'
. . .
(TS) % UCO-000.093803  %  NBR0706 TEST OUTPUT FOR CONSOLE 'TS'
 <* ? TSC-000.093804 NSI0077 ENTER AUTOMATIC COMMAND FILE NAME. REPLY (FILE NAME; N(DO NOT USE); EOT(USE STANDARD))
 TSC R(CO)-000.093917. ---------------------------------------------------------------------------------------    (13)
  <* % UCO-000.093917  %  NBR0792 QUESTION '000' FROM ' TSC' ANSWERED BY '(CO)'. REPLY:
  <* ?   Q-000.093918  %  NSI6005 SYSTEM PARAMETER STUPTYPE = C. SHALL VALUE BE CHANGED? REPLY ( U(NCHANGED), W(ARM),
                          C(OLD), S(ELECTIVE), Z(IP), T(TSN FILE RESET ONLY), J(JOIN AND TSN FILE RESET),EOT=UNCHANGED)
   Q R(CO)-000.093930.C --------------------------------------------------------------------------------------    (14)
  <* ?   Q-000.093930  %  NSI6010 SYSTEM PARAMETER RECONUC = N. SHALL VALUE BE CHANGED? REPLY ( U(NCHANGED), N(O),
                          B(ACKUP), T(SOSCAT), A(LL), R(ESET), EOT=UNCHANGED
  <* % UCO-000.093930  %  NBR0792 QUESTION '000' FROM '   Q' ANSWERED BY '(CO)'. REPLY: C
   Q R(CO)-000.093936. ---------------------------------------------------------------------------------------    (15)
  <* % UCO-000.093936  %  NBR0792 QUESTION '000' FROM '   Q' ANSWERED BY '(CO)'. REPLY:
  <* %   Q-000.093936  %  DMS035B IMCAT TASK 'XAA9' FOR PUBSET WITH PUBSET ID 'FCH1' CREATED AND STARTED
  <* %XAA9-000.093940  %  BLP0990 PROCESSING OF REP FILE ':FCH1:$TSOS.SYSREP.DSSM.043' STARTED
  <* %DSSM-000.093941  %  BLP0995 NOREF FILE IS ':FCH1:$TSOS.SYSNRF.DSSM.043'
  <R %DSSM-000.093942  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'CALENDAR/V18.0' COMPLETELY PROCESSED
  <R %DSSM-000.093942  %  ESM0220 FUNCTION 'CREATE' FOR SUBSYSTEM 'MSCFANC /V18.0' COMPLETELY PROCESSED
. . .
  <* %   M-000.093946  %  NMH1102 MESSAGE OUTPUT FILE ':FCH1:$TSOS.SYSMES.BS2CP.180', ACCESS=DLAM + ISAM,
                          ACTION=STARTUP
  <* %   M-000.093946  %  NMH1102 MESSAGE OUTPUT FILE ':FCH1:$TSOS.SYSMES.EKP.01', ACCESS=ISAM, ACTION=STARTUP
  <* %   M-000.093946  %  NMH1112 MESSAGE PROCESSING READY
. . .
  <* %SERS-000.094039  %  NER1500 SERSLOG LOGGING FILE ':FCH1:$TSOS.SYS.SERSLOG.2011-11-10.013.01' OPENED
  <H % TSC-000.094049  %  HEL0001 HW ERROR LOGGING FILE ':FCH1:$TSOS.SYS.HEL.2011-11-10.091614' OPENED
                          WITH 'SPACE=  402'
  <R % TSC-000.094049  %  EXC0040 LOGGING FILE ':FCH1:$SYSAUDIT.SYS.REPLOG.2011-11-10.013.01' OPENED
  <@ %HT1E-000.094052  %  CMD0695 SYSTEM SYNTAX FILE ':FCH1:$TSOS.SYSSDF.SDF.047' ACTIVATED
  <@ %HT1E-000.094052  %  CMD0695 SYSTEM SYNTAX FILE ':FCH1:$TSOS.SYSSDF.ACO.022' ACTIVATED
. . .
  <* %DER1-000.094128  %  NJA0003 D.E.R. server task DER1 started
  <A % TSC-000.094128  %  NJA0001 D.E.R INITIALIZED SUCCESSFULLY
  <* % TSC-000.094128  %  NAM0015 STARTUP CONTINUES WITHOUT ACCOUNTING
  <* % MSG-000.094128  %  ETMRK18 CPU 01 ATTACHED
  <* % TSC-000.094129  %  NSI0000  ***  S Y S T E M   R E A D Y  *** ---------------------------------------    (16)
  <J % TSC-000.094129  %  JMS0066 JOB 'JSSTD' ACCEPTED ON 11-11-10 AT 09:41, TSN = 20ZE
  <J % TSC-000.094129  %  JMS0066 JOB 'JSSTD1' ACCEPTED ON 11-11-10 AT 09:41, TSN = 20ZF
  <J % TSC-000.094129  %  JMS0066 JOB 'JSSTD2' ACCEPTED ON 11-11-10 AT 09:41, TSN = 20ZG
  <J % TSC-000.094129  %  JMS0066 JOB 'JSTSOS' ACCEPTED ON 11-11-10 AT 09:41, TSN = 20ZH
OPRR /(CO)-000.094129 RUN CMDFILE ------------------------------------------------------------------------    (17)
  <* % UCO-000.094129  %  NBR0972 OPERATOR TASK WITH TSN 'XABB' CREATED FOR PROCESSING OF /RUN COMMAND SEQUENCES
  <R % TSC-000.094129  %  JMS0300 JOB STREAM '$SYSJS' 'ATTACHED'
(CO) +RUNT-000.094129  %  NBR1000 COMMAND '/RUN' RECEIVED. RUN ID '0001' ASSIGNED
(CO) +RUNT-000.094129  %  NBR1001 RUN ID '0001'. READING OF /RUN COMMAND FILE ':FCH1:$TSOS.CMDFILE' STARTED
```

(1) Der DIALOG-Startup wird ohne zusätzliche Option (Antwort: `EOT`) weitergeführt.

(2) Der Home-Pubset ist online.

(3) Datum und Uhrzeit laut SVP werden angezeigt.

(4) Die SYSSTART-Rep-Datei wird verarbeitet.

(5) Es soll die Standard-Startup-Parameterdatei verarbeitet werden (Antwort: `EOT`).

(6) Es soll keine weitere Parameterdatei verarbeitet werden (Antwort: `END`).

(7) Der Dateiname für BS2000/OSD soll eingegeben werden. Mit der Antwort `EOT` wird BS2000/OSD aus der Standarddatei geladen.

(8) Mit der Antwort `EOT` wird die Standard-Rep-Datei verarbeitet.

(9) Es soll keine weitere Rep-Datei verarbeitet werden (Antwort: `END`).

(10) Es werden verschiedene Systeminformationen ausgegeben: Name und Version des Betriebssystems, Zeitpunkt der Generierung, verfügbarer Speicherplatz, Zeitzoneninformationen, TODR-Epoche, Konfigurationsdaten, Größe der Seitenwechseldateien.

(11) Der seitenwechselbare Teil des BS2000-Ablaufteils wurde erfolgreich geladen und korrigiert (REP-Verarbeitung). Er ist somit ablaufbereit.

(12) Die unter (6) ausgewählten Parametersätze werden in der CONSLOG protokolliert. Die Verarbeitung findet – je nach Parameter-Schlüsselwort – zu unterschiedlichen Zeitpunkten statt.

(13) Die Kommandodatei mit dem durch den Systemparameter CMDFILE festgelegten Namen soll nach „System Ready“ automatisch verarbeitet werden (Antwort: `EOT`). Wird mit `N` geantwortet (`DO NOT USE`), kann eine CMDFILE später mit dem Kommando RUN CMDFILE nachträglich gestartet werden.

(14) Es soll ein Kaltstart durchgeführt werden (Antwort: `C`).

(15) Die Katalog-Rekonfiguration soll wie im Parameterservice definiert erfolgen (Antwort: `N`). Im Beispiel also mit RECONUC=N.

(16) „System Ready“, das BS2000-System ist betriebsbereit.

(17) Automatisches Starten der Kommandodatei `CMDFILE` (siehe (14)).

# 2.3 Systemkorrekturen

## 2.3.1 Funktion und Aufbau einer Rep-Datei

In der Systemeinleitung können zur Korrektur der Ladeobjekte SYSIPL, SYSSTART, SLED, und SYSFIRST, sowie zur Korrektur des Organisationsprogramms Rep-Dateien, bestehend aus Rep-Sätzen, verwendet werden.

Rep-Sätze erlauben eine byteweise Korrektur der oben angeführten Ladeobjekte. Auch sog. „Selectable Units“ (entkoppelte Liefereinheiten), die nicht zum Exec gebunden sind, können über Rep-Sätze korrigiert werden. Die entsprechende Rep-Datei ist mit dem Namen SYSREP.SU<entryname> unter TSOS katalogisiert. Ganze Module können nicht über Rep-Sätze ausgetauscht werden, sondern müssen mit LMS in die dazugehörige Bibliothek (OML) eingebracht werden.

Die Repverarbeitung findet in der Regel für das zuletzt geladene Objekt statt. Das Laden des Betriebssystems BS2000/OSD läuft in zwei Schritten ab; die beiden Teile werden einzeln korrigiert.
Klasse-1-Rep-Sätze werden unmittelbar nach dem Laden des residenten Teils des Organisationsprogramms verarbeitet. Sie sollen nur für diejenigen Klasse-1-Module (residenter Teil) des Organisationsprogramms erstellt werden, welche zum Laden und Initialisieren der Klasse-2-Module (nichtresidenter, seitenwechselbarer Teil) des Organisationsprogramms benötigt werden. Die Relativierung der Korrekturdaten in Klasse-1-Rep-Sätzen ist auf Klasse-1-Module und Module der Systemeinleitung beschränkt; Entries können nur zum Modulanfang relativiert werden.
Klasse-2-Rep-Sätze werden unmittelbar nach dem Laden des nichtresidenten Teils des Organisationsprogramms verarbeitet. Mit ihnen können alle Module des Organisationsprogramms korrigiert werden. Die Relativierung von Korrekturdaten ist hier für alle Module und Entries möglich.

### Rep-Verarbeitung bei der Systemeinleitung

Die Rep-Verarbeitung wird bei FAST- und AUTOMATIC-Startup mit Ausnahme von Fehlerfällen automatisch durchgeführt, d.h. ohne Dialog mit dem Operator. Beim DIALOG-Startup kann die Rep-Verarbeitung, außer für SYSIPL und SLED, vom Operator beeinflusst werden.

Beim FAST- und AUTOMATIC-Startup werden die zu verarbeitenden Reps in den Dateien $TSOS.SYSREP.BS2.180 und $TSOS.SYSREP.STRT.180 auf dem Home-Pubset erwartet, sofern über die Startup-Parameterdatei keine anderen Rep-Dateien für das Organisationsprogramm angegeben werden (Parameter REPFILEx).
Ein Dialog mit dem Operator wird nur in Fehlerfällen aufgenommen.

Die Standardnamen der Rep-Dateien lauten:

| | | |
|---|---|---|
| für BS2000/OSD: | SYSREP.BS2.180 | |
| für SYSIPL: | SYSREP.IPL.180 [1] | kopiert in Datei SYSREP.IPL.DSKnnn |
| für SYSSTART: | SYSREP.STRT.180 | |
| für SLED: | SYSREP.SLED.180 [1] | kopiert in Datei SYSREP.SLED.DSKnnn |

[1] Diese Dateien werden durch SIR kopiert und dann im SVL verankert. Sie können nicht über Dateinamen angesprochen werden (siehe unten).

Beim DIALOG-Startup können Rep-Dateien auf gemeinschaftlicher Platte oder Privatplatte vorliegen oder über Bedienstation eingegeben werden.

Die Reihenfolge der Verarbeitung wird vom Operator festgelegt, indem er auf die Meldung `NSI0050` jeweils ein Eingabegerät angibt. Die Reps werden sofort verarbeitet. Danach erscheint wieder die Meldung `NSI0050`. Dieser Vorgang wiederholt sich solange, bis der Operator als Antwort `P.END` (bzw. `P.`, wenn bereits eine Repdatei mit Standardnamen verarbeitet wurde) eingibt (siehe folgende Seite).

Eine Plattendatei kann viermal und die Bedienstation zweimal als Eingabe angegeben werden. Es wird überprüft, ob die angegebene Datei bereits einmal verarbeitet wurde. Diese Einschränkungen gelten nur für die BS2000-Rep-Dateien; für alle anderen Objekte gibt es diese Einschränkungen nicht.

Der Dialog an der Bedienstation für Klasse-1-Rep-Sätze und Klasse-2-Rep-Sätze wird separat eröffnet. Die eingegebenen Daten werden genauso behandelt, als wenn sie über eine Rep-Datei eingegeben werden.
Sie werden also auch in die Sicherstellungsdatei SYS.NSI.SAVEREP und später in die Datei REPLOG ($SYSAUDIT.SYS.REPLOG.<datum>.<session-nr>.01) geschrieben.



*Rep-Verarbeitung für SYSIPL und SLED*

Reps für SYSIPL und SLED befinden sich jeweils in einer Rep-Datei, die von SIR im SVL verankert wird. Der Operator hat während der Systemeinleitung bzw. der Erstellung von Diagnoseinformationen keine Möglichkeit, eine andere Rep-Datei auszuwählen.
Aus demselben Grund hat auch die Art der Systemeinleitung keinen Einfluss auf diesen Teil der Rep-Verarbeitung.
Bei einer Änderung dieser Rep-Dateien muss erneut die Verankerung im SVL mit SIR (Funktion CREATE-IPL oder MODIFY-IPL) erfolgen.

### Aufbau einer Rep-Datei

Eine Rep-Datei für die Systemeinleitung hat folgenden Aufbau:

| Satz | Beschreibung |
|---|---|
| BS2000_LOADER[_kommentar] | 1. Satz (obligatorisch) |
| Klasse-1-Rep-Sätze | Rep-Sätze für ausgewählte Module des residenten Teils des Organisationsprogramms und der Systemeinleitung (wahlweise) |
| _END[_kommentar] | Ende-Anweisung für Klasse-1-Rep-Sätze (obligatorisch) |
| Klasse-2-Rep-Sätze | Rep-Sätze für das gesamte Organisationsprogramm (wahlweise) |
| _END[_kommentar] | Ende-Anweisung für Klasse-2-Rep-Sätze |
| /[_kommentar] | Dateiende-Kennung |
| *%kommentar<br>oder<br>*%%kommentar | Kommentarsatz wird an der Bedienstation ausgegeben (gilt nicht für Kommentare in SYSREP.IPL.vvv) |

Dieser Aufbau gilt für SYSIPL, SLED, SYSSTART, BS2000 und SYSFIRST gleichermaßen. Die Unterscheidung zwischen Klasse-1-Repverarbeitung und Klasse-2-Repverarbeitung ist nur für BS2000/OSD relevant.

Als Endekriterium für Klasse-2-Repsätze und die Rep-Datei muss entweder eine zweite END-Anweisung oder eine Dateiendekennung existieren.

Auf Platte können die Rep-Sätze 1-256 Byte lang sein, wobei die Zeichen nach dem 80. Byte nicht verarbeitet werden.
Die Rep-Datei ist eine SAM-Datei mit variablen Sätzen und Standardblockung BUFFER-LENGTH=STD(SIZE=1) oder (SIZE=2). Der Dateiname kann beliebig gewählt werden. Änderungen in der Rep-Datei auf Platte sollten nur mit RMS (siehe Abschnitt „RMS: Montage und Lieferung von Reps“ auf Seite 70) durchgeführt werden.

Die Rep-Dateien werden in der Reihenfolge gelesen und verarbeitet, die von der Systembetreuung festgelegt wird.

Die Bedienstation kann zweimal als Rep-Eingabegerät angegeben werden. Wurden evtl. fehlerhafte Platten-Reps eingelesen, so können diese zum Schluss noch einmal über Bedienstation korrigiert werden.
Kommentarsätze (mit * in Spalte 1) können nach dem Satz „BS2000 LOADER“ an beliebiger Stelle der Rep-Datei eingefügt werden. Diese werden vom Startup nicht ausgewertet. Kommentarsätze, die in der 2. Spalte das %-Zeichen enthalten, werden über die Bedienstation protokolliert (gilt nicht für Kommentare in SYSREP.IPL.vvv).

Rep-Sätze für Module, die nicht in das Organisationsprogramm eingebunden sind, deren Namen dem Organisationsprogramm jedoch bekannt sind, werden ohne Fehlermeldung übergangen. Damit ist es möglich, Rep-Sätze für alle Module einer BS2000-Version in eine Rep-Datei zu integrieren.

Rep-Sätze, die einen ungültigen Modulnamen enthalten, werden als fehlerhaft protokolliert. Enthält der Rep-Satz jedoch in Spalte 69 ein „S“ oder „U“, dann wird die Fehlermeldung unterdrückt. Damit können Rep-Sätze für Module in die Rep-Datei aufgenommen werden, die dem Organisationsprogramm (noch) unbekannt sind (z.B. Selectable Units, Seite 60).

## 2.3.2 Rep-Sätze

Die Adresse der zu korrigierenden Daten ist stets relativ zum Modulanfang anzugeben. Jeder Rep-Satz wird vor der Verarbeitung auf richtiges Format untersucht. Prüfdaten, Kontrollzahl und Modulversion werden nur geprüft, wenn sie angegeben sind, d.h. zu Testzwecken können sie auch weggelassen werden. Fehlerhafte Sätze werden zusammen mit einer Fehlermeldung an der Bedienstation protokolliert, eine Korrektur wird für solche Sätze nicht ausgeführt.

Korrekturdaten, die von der Adresse eines anderen Moduls abhängen, können in der Form „Basis + Distanz“ angegeben werden. Diese Art Reps werden „relative Reps“ genannt. Die Distanz ist relativ zum Modulanfang/Entry/ISL-Entry anzugeben und die Basis in Gestalt des Modul-/Entry-/ISL-Entry-Namens. Damit sind Rep-Sätze invariant gegenüber Änderungen in allen übrigen Modulen des Objektes.

In der folgenden Tabelle steht:

- **a** für ein alphanumerisches Zeichen (0-9, A-Z)
- **x** für ein Sedezimalzeichen (0-9, A-F)

| **Spalte** | **Inhalt** | **Bedeutung** |
|---|---|---|
| 1 | _ | Leerzeichen |
| 2 - 4 | REP | |
| 5 | _ | Leerzeichen |
| 6 -10 | xxxxx | Rep-Adresse, relativ zum Modulanfang |
| 11 | _ | Leerzeichen |
| 12 -14 | 3-stellige Zahl | Laufende Nummer der Objektkorrekturversion |
| 15 | _ | Leerzeichen |
| 16 | X, I, O, S, P, T | X: Standard- oder relativer Rep<br>I: relativer Rep (ISL-Entry)<br>O, S, P, T: relativer Rep für X86-64-Code |

Tabelle 2: Format der Rep-Sätze (Teil 1 von 2)

| **Spalte** | **Inhalt** | **Bedeutung** |
|---|---|---|
| 17 - max. 50 | ' xxx...x'<br>oder<br>' xx...x' + Name | max.32 Korrekturdaten, in Anführungszeichen eingeschlossen,<br>oder<br>max. 22 Korrekturdaten, in Anführungszeichen eingeschlossen, gefolgt von einem Pluszeichen und dem Namen eines Moduls/ Entries/ISL-Entries. Die Adresse wird von der Systemeinleitung zu den letzten 8 Stellen der Korrekturdaten addiert. Der Name muss (wie in Spalte 73-80) 8-stellig angegeben werden.<br>X86-64-Reps haben ein gesondertes Rep-Format (siehe Seite 65) |
| 51 | _ | Leerzeichen |
| 52 - 55 | xx__ oder xxxx | 2 oder 4 Prüfdaten; es sind das erste oder die ersten beiden Byte anzugeben, die von den Korrekturdaten überschrieben werden sollen. |
| 56 | _ | Leerzeichen |
| 57 | x | Kontrollzahl (parity digit) für Rep-Adresse, Korrekturdaten und Prüfdaten |
| 58 - 65 | aaaaaaaa | Nummer der Problemmeldung |
| 66 - 68 | aaa | Modulversion |
| 69 | a | Rep-Kennzeichen |
| 70 | 1 oder 2 | 1 für Klasse-1-Rep-Sätze, 2 für Klasse-2-Rep-Sätze |
| 71 | a oder Leerzeichen | Laderversion (A-Z) |
| 72 | a | Kennzeichen für die Auswahl von Rep-Sätzen für unterschiedliche Code-Varianten.<br>'K' kennzeichnet Korrekturen von X86-64-Code.<br>Ein Leerzeichen kennzeichnet Korrekturen für /390-Code. |
| 73 - 80 | aaaaaaaa | Modulname, 8-stellig; die Anfangsadresse dieses Moduls wird zu der Rep-Adresse addiert. |

Tabelle 2: Format der Rep-Sätze (Teil 2 von 2)

*Hinweise zum Format der Rep-Sätze*

- Klasse-1-Rep-Sätze und Klasse-2-Rep-Sätze besitzen dasselbe Format und erlauben dieselben Korrekturfunktionen. Sie unterscheiden sich lediglich durch die Klassenangabe in Spalte 70 und ihre Verwendung sowie die Möglichkeit zur Angabe von Entries/ISL-Entries bei „relativen Reps“.
- Die Adresse der zu korrigierenden Daten wird stets durch Addition der Moduladresse (Name in Spalte 73ff) und der Rep-Adresse gebildet.
- Um bei relativen Reps Unterscheidungsmöglichkeiten zu haben, werden zusätzlich zum Format mit dem Indikator `X` die Indikatoren `I, O, S, P`, und `T` angeboten. Die Rep-Daten werden im Anschluss an den Indikator in einschließenden Hochkommas angegeben.
  - Indikator `X` bezeichnet reale Module/Entries. Die Indikatoren `P` und `T` kennzeichnen spezielle relative Reps für X86-64-Code. Dazu gibt es ein spezielles Format:

    Format: `X`'distanz'+basisadresse
    Hier wird für die Basisadresse die Adresse des realen Entrys vermerkt.
  - Indikator `I` zeigt an, dass der angegebene Entry-Name den ISL-Entry bezeichnet.

    Beispiel: I'distanz'+<name-des-isl-entrys>
  - Die Indikatoren `O` und `S` kennzeichnen spezielle relative Reps für X86-64-Code. Dazu gibt es ein spezielles Format:

    Beispiel: `O'<code>','<distanz>'+<entry/module-name>`
    Aus <code> und berechneter Adresse wird die Korrekturinformation aufgebaut. <code> und <distanz> sind jeweils genau 8 Zeichen lang.
- Da die Korrekturdaten in hexadezimaler Schreibweise angegeben werden, muss deren Anzahl stets gerade sein.
- Bei zu relativierenden Korrekturdaten müssen mind. 8 Korrekturdaten vorhanden sein.
- Falls die zu korrigierenden Daten eine Adresse darstellen, ist der alte Wert generierungsabhängig, d.h. es sollten keine Prüfdaten angegeben werden.
- Die Kontrollzahl (Spalte 57) dient zur Sicherung des Rep-Satz-Inhalts. Sie ergibt sich aus der Summe aller Stellen von Rep-Adresse, Korrekturdaten und Prüfdaten
- Zur Summe ist die Anzahl der Korrekturdaten zu addieren. Das Ergebnis mod(16) ergibt für die Kontrollzahl einen Wert zwischen 0 und F.
- mod(16) bedeutet: Die Summe ist durch 16 zu dividieren. Der verbleibende Rest ist die gesuchte Kontrollzahl. Die Kontrollzahl wird mit dem Dienstprogramm RMS (Rep Montage System) automatisch erzeugt, wenn sie nicht vorhanden ist.
- Der Inhalt der Spalten 58-65 dient organisatorischen Zwecken und wird von der Systemeinleitung nicht ausgewertet.

- Wahlweise können Prüfdaten, Kontrollzahl und Modulversion weggelassen werden. Bei Startup entfallen dann die entsprechenden Prüfungen.
- Folgende Rep-Kennzeichen in Spalte 69 sind definiert:

| | | |
|---|---|---|
| D | = | Diagnose-/Fang-Rep |
| O | = | Optionaler Rep |
| Q | = | Selectable Unit, Diagnose-/Fang-Rep |
| S | = | Selectable Unit |
| T | = | Trace (einschalten) |
| U | = | Selectable Unit, optional |
| V | = | Vorläufiger Rep |
| _ | | Normaler Rep |

- Das Kennzeichen in der Spalte 72 steuert die Auswahl der Rep-Sätze für unterschiedliche Code-Varianten (/390- oder X86-64-Code). Bei Kennzeichnung mit 'K' wird die Korrektur nur für X86-64-Code berücksichtigt. Enthält die Spalte 72 ein Leerzeichen, gilt die Korrektur für /390-Code. Alle anderen Inhalte werden ignoriert.
  Entsprechend dem Ablauf auf den BS2000/OSD-Servern werden die Reps für das jeweils andere HSI ignoriert.
- Anstelle eines Modulnamens kann bei Klasse-2-Rep-Sätzen in den Spalten 18-49 auch ein ENTRY-Name stehen.
- Anstelle eines Modulnamens kann in den Spalten 73-80 auch ein ENTRY- oder CSECT-Name stehen. In diesen Fällen darf dann keine Modulversion angegeben sein (nur für Klasse-2-Rep-Sätze).

### Eingabe von Rep-Sätzen über Bedienstation

Während der Ladeprozedur erscheint folgende Meldung:

```
?P.NSI0050 SPECIFY BS2000 REP FILE OR DEVICE. REPLY (EOT (USE STANDARD
           FILE); FN=FILENAME(VOL=VSN); CONS; END)
```

Antwortet der Operator mit `P.CONS` beginnt der Dialog für die Klasse-1-Rep-Sätze:

Meldung:

```
?P.NSI0070 ENTER CLASS 1 MODULE NAME. REPLY (NAME; EOT; (NO MORE CLASS 1
           CONSOLE REPS))
```

Antwort: `P.<modulname>`
(Name des Moduls, das geändert werden soll)

Meldung:

```
?P.NSI0071 ENTER RELATIVE ADDRESS IN MODULE. REPLY (5 CHAR)
```

Antwort: `P.<korrekturadresse>`
(Modul-relative Adresse eines Reps; 5 sedezimale Ziffern)

Meldung:

```
?P.NSI0073 ENTER CORRECTION DATA. REPLY (MAX 32 CHARACTERS)
```

Antwort: `P.<änderungsdaten>`
(Daten zur Korrektur eines Modulobjektcodes, 2-32 sedezimale Ziffern)

Die Eingabe von relativen Reps ist erweitert:
Neben der Eingabe von distanz+basisadresse ist jetzt auch `X'distanz'+basisadresse` und `I'distanz'+<isl-entry-adresse>` sowie die Eingabe der entsprechenden relativen Repformate für X86-64-Code möglich.
`basisadresse` ist die Adresse eines realen Entrys oder Moduls.

Meldung:

```
?P.NSI0074 ENTER CHECK INFORMATION. REPLY (CCCC,P,MMM (OLD DATA, PARITY,
           VERSION); EOT(SKIP CHECK))
```

mögliche Antworten:

| | |
|---|---|
| CCCC | 1 oder 2 Prüfbytes |
| P | Kontrollzahl oder leer |
| MMM | Modulversionsnummer oder leer |

Für alle eingegebenen Daten findet eine Prüfung statt, bei fehlenden Daten wird die entsprechende Prüfung nicht durchgeführt.

oder Antwort: `P.` (und ENTER-Taste drücken)
Bei dieser Antwort findet keine Prüfung statt.

Danach kommt wieder die Meldung `NSI0070` und fordert die Eingabe des nächsten Rep-Satzes an. Der gleiche Dialog wiederholt sich solange, bis keine Klasse-1-Rep-Sätze mehr eingegeben werden sollen. In dem Fall muss die Meldung `NSI0070` mit `P.` und dem Drücken der ENTER-Taste beantwortet werden.

Nach weiteren informativen Meldungen beginnt der Dialog für Klasse-2-Rep-Sätze. Er wird eingeleitet mit der Meldung

```
?P.NSI0075 CONSOLE ASSIGNED AS REP-LOADER FOR CLASS2 MODULES.
           REPLY (EOT(CONSOLE IS USED); N(NO)).
```

Die Antwort darauf ist `P.` oder `P.N` (keine Klasse-2-Rep-Sätze) und ENTER-Taste drücken.

## 2.3.3 Rep-Zwischenspeicherung

Alle Rep-Sätze werden in der Datei SYS.NSI.SAVEREP zwischengespeichert und dann in der REPLOG-Datei protokolliert. Falls keine aktuelle REPLOG-Datei (siehe Seite 69) vorhanden ist, wird die SAVEREP-Datei von SLED sichergestellt und kann wahlweise mit DAMP aufbereitet werden.

**SAVEREP (Systemeinleitung für BS2000/OSD)**

Jeder Rep-Satz, der das System tatsächlich modifiziert hat, wird mit einer Ziffer in der Spalte 72 gekennzeichnet, die anderen Rep-Sätze erhalten einen Buchstaben.
Dabei bedeutet:

| | |
|---|---|
| 1 oder A | Rep von Platte |
| 4 oder D | Rep von Bedienstation |

Der Lader-Code wird in den Satz BS2000 LOADER geschrieben. Der Lader-Code wird als Funktion aller Rep-Sätze (ohne SYSIPL-Reps) erzeugt und dient zur Identifizierung der Rep-Datei. Er ergibt sich aus der Summe der Rep-Adressen und der Kontrollzahlen aller existierenden Rep-Sätze.

Für jedes Rep-Medium werden bei der Sicherstellung je zwei START- und END-Sätze eingetragen, die Datum, Zeit, Komponente und den vollständigen Dateinamen zur eindeutigen Identifizierung enthalten.
Reps für ein anderes als das aktuelle HSI werden nicht in der SAVEREP protokolliert (und damit auch nicht in der REPLOG) und auch bei der Meldung zur Verarbeitung der IPL-Reps nicht mitgezählt.

Diese Sätze werden in folgender Form eingetragen:

```
START <objekt> yyyy-mm-tt,hh:mm:ss
START <datei oder eingabegerät>
:
: <Rep- und Kommentarsätze>
:
END <objekt> yyyy-mm-tt,hh:mm:ss
END <datei oder eingabegerät>
```

Für <objekt> können folgende Werte stehen:

| | |
|---|---|
| SYSIPL: | Reps für SYSIPL |
| SYSSTART: | Reps für SYSSTART |
| BS2000-CL1: | Reps für Klasse-1-Exec |
| BS2000-CL2: | Reps für Klasse-2-Exec (ohne Datum und Uhrzeit) |
| VM2000: | Reps für VM2000-Hypervisor (S-Server) |

<datei> enthält den vollständigen Dateinamen der Rep-Datei aus den Teilen Katalogkennung, Benutzerkennung und Dateiname.

Für <eingabegerät> können folgende Werte stehen:

| | |
|---|---|
| IPL-Pfad | für SYSIPL-Reps |
| *CONSOLE | für Reps, die von der Bedienstation eingegeben werden |

Nach dem letzten END-Satz für BS2000-Klasse-2-Reps wird ein zusätzlicher Satz (END BS2000-Satz) hinterlegt, der Informationen über die Anzahl der verarbeiteten und ersetzten Reps enthält:

```
END BS2000: LOADER VERSION Z, mmmmm PROCESSED REPS, nnnnn REPLACED REPS
```

Die Datei SYS.NSI.SAVEREP wird im weiteren Verlauf der Systemeinleitung in die Logging-Datei $SYSAUDIT.SYS.REPLOG.<datum>.<session-nr>.01 kopiert. Dort werden die Korrekturdaten für BS2000/OSD und alle nachgeladenen Systemteile (Subsysteme) protokolliert.

### REPLOG

Die Korrekturdaten für SYSIPL, SYSSTART, BS2000 und alle nachgeladenen Subsysteme sowie alle mit ROSI im laufenden System durchgeführten Korrekturen werden in der Loggingdatei $SYSAUDIT.SYS.REPLOG.<datum>.<session-nr.>.01 protokolliert.



Zusätzlich wird, neben den tatsächlich durchgeführten Korrekturen von BS2000/OSD und den nachgeladenen Systemteilen/Subsystemen, die alte Information, welche durch die Korrektur ersetzt wird (Rep before image), im Speicher protokolliert. Dadurch können Korrekturen zuverlässig und konsistent erfasst und aus dem laufenden System wieder entfernt werden.

Mit dem Systemparameter SECSTART kann die Systembetreuung festlegen, ob der Korrekturstand über mehrere Systemläufe hinweg protokolliert werden soll.
Bei SECSTART=N werden alle evtl. existierenden REPLOG-Dateien gelöscht.
Bei SECSTART=Y bleiben die REPLOG-Dateien erhalten; die Korrekturen können über mehrere Systemläufe hinweg verfolgt werden.

Mit dem Kommando SET-REPLOG-READ-MARK kann die Systembetreuung die REPLOG-Datei des aktuellen Systemlaufs kurzzeitig schließen. Alle bis zu diesem Zeitpunkt protokollierten Daten können so kopiert und analysiert werden.

| Kommando | Bedeutung |
|---|---|
| SET-REPLOG-READ-MARK | Lesenden Zugriff auf die REPLOG-Datei ermöglichen |

### 2.3.4 RMS: Montage und Lieferung von Reps

Das Dienstprogramm RMS (Rep Montage System) ist ein dialogorientiertes Programm für die Montage und Lieferung von Rep-Mengen.

Das Programm verwaltet in einer zentralen Datei komprimiert alle Korrekturen und Beschreibungen, deren Herkunft und Produktzugehörigkeit sowie Umfang und Zeitpunkt aller Aktionen. Es stehen somit alle für die Lieferung und den Einsatz der Korrekturen relevanten Informationen zur Verfügung und jede Aktion ist für die Fehlerdiagnose beliebig reproduzierbar.

RMS ist im Handbuch „Dienstprogramme“ [15] beschrieben.

## 2.4 Systembeendigung

Es gibt zweierlei Arten der Systembeendigung:

- Planmäßige bzw. normale Beendigungen sind alle Beendigungen, die durch das SHUTDOWN-Kommando eingeleitet werden.
- Außerplanmäßige bzw. abnormale Beendigungen können als Folge von Software- oder Hardwarefehlern auftreten.

| **Kommando** | **Bedeutung** |
|---|---|
| BCEND | DCM im Server beenden |
| INFORM-ALL-JOBS | Nachricht an alle Benutzertasks senden |
| SET-RESTART-OPTIONS | automatischen Restart steuern |
| SHOW-RESTART-OPTIONS | Informationen über automatischen Restart anfordern |
| SHUTDOWN | Systemlauf beenden |
| UNLOCK-DISK | Systembelegungsprotokoll bereinigen |

Tabelle 3: Kommandoübersicht zur Systembeendigung

## 2.4.1 Planmäßige Beendigung

Bevor der Operator das System endgültig beendigt, sollte er den Dialogteilnehmern mit dem Kommando SHUTDOWN MODE=QUIET eine Warnung zukommen lassen, um ihnen die Gelegenheit zu geben, ihre Tasks ordnungsgemäß abzuschließen.

Dabei wird zum einen ein INFORM-ALL-JOBS-Kommando simuliert zur Benachrichtigung des Teilnehmerbetriebes, zum anderen wird ein BCEND W=Y,TERM=N abgesetzt, der den Teilhaberanwendungen eine Warnung über den bevorstehenden BCEND zukommen lässt. Ab diesem Zeitpunkt werden keine weiteren Benutzer zum System zugelassen und keine neuen Jobs (z.B. mit ENTER-JOB) gestartet.

Die endgültige Systembeendigung leitet der Operator mit dem Kommando SHUTDOWN MODE=END ein. Alle noch laufenden Tasks werden abgebrochen, und das Kommando BCEND W=N wird simuliert. SPOOL- und Job-Einträge werden jedoch sichergestellt und, je nach Art des Systemstarts, im nächsten Systemlauf evtl. wieder abgearbeitet.
Alle Privatplattenbelegungen werden abgebaut.

Es gibt die Möglichkeit, mit dem Kommando SHUTDOWN MODE=END(RESTART=*YES) einen automatischen Restart nach Systembeendigung einzuleiten.
Mit SHUTDOWN MODE=END(RESTART=*YES(DELAY=...)) kann dieser Restart zeitverzögert eingeleitet werden. Beide Erweiterungen sind bei der Fernbedienung des Systems von Relevanz.

Mit dem Kommando SHUTDOWN MODE=END(RESTART=*YES(IPL-DEVICE=...)) kann ein automatischer Restart mit Wechsel der IPL-Platte durchgeführt werden.
Mit dem Wechsel der IPL-Platte ist auch der Restart eines anderen Systems (im Modus AUTOMATIC) möglich.

*Systembeendigung unter VM2000*

Für Gastsysteme unter VM2000 wird auch das Kommando SHUTDOWN-VM zur Systembeendigung durch den Administrator angeboten.
Für ausführliche Informationen siehe Handbuch „VM2000“ [62].

*Systembeendigung über X2000 auf SQ-Servern*

Der Shutdown von BS2000/OSD kann auch über den SQ-Manager im Menü *Server Unit* > *BS2000-Bedienung* > *Bedienung* mit der Funktion „BS2000 Shutdown durchführen“ eingeleitet werden. Dieser Shutdown wird durch X2000 immer ohne Zeitbegrenzung eingeleitet. Die Shutdown-Anforderung wird in einer dynamisch kreierten System-Task ausgeführt und an der Konsole protokolliert mit `NRT1201 SHUTDOWN REQUESTED BY 'X2000'`.

Ein Shutdown von BS2000/OSD wird im SQ-Manager auch im Menü *Server Unit* > *Server* > *Administration* > *Ein/Aus* mit den Funktionen „Server Unit sofort neu starten (Reboot)“ und „Server Unit sofort ausschalten“ eingeleitet. Dabei bleibt dem BS2000-System nur die ggf. im SQ-Manager (Menü *Server Unit* > *BS2000-Bedienung* > *Optionen*) eingestellte Restlaufzeit für den ordnungsgemäßen Shutdown.

Die Shutdown-Anforderung wird in einer dynamisch kreierten System-Task ausgeführt. Sie wird bei eingestellter Restlaufzeit an der Konsole protokolliert mit `NRT1200 SHUTDOWN WITH RUNTIME LIMIT REQUESTED BY 'X2000'. REMAINING RUNTIME: '<HH>:<MM>'`. Wenn keine Restlaufzeit eingestellt ist, dann wird, wie oben, die Meldung `NRT1201` ausgegeben.

Die Restlaufzeit wird in der Jobvariablen `$TSOS.REMAINING-BS2000-RUNTIME` abgelegt und mit der ablaufenden Zeit heruntergezählt. Die Jobvariable besteht aus fünf Zeichen mit den möglichen Werten:

– `UNDEF` keine Shutdown-Anforderung durch X2000

– `UMLIM` keine Zeitbegrenzung

– `<hh>:<mm>`
verbleibende Zeit bis zum abrupten Anhalten von BS2000/OSD durch X2000

Über den Systemparameter SHUTPROC (siehe Seite 758) kann zusätzlich gesteuert werden, ob BS2000/OSD nach einer Anforderung durch X2000 sofort heruntergefahren oder ob zur Systembeendigung ein Enter-Job gestartet werden soll.

Das sofortige Herunterfahren erfolgt analog dem Kommando
SHUTDOWN MODE=*END(RESTART=*NO).

Der Enter-Job wird ggf. in der Datei $TSOS.SYSENT.SHUTDOWN erwartet. Er wird unter der Benutzerkennung, die beim Kommando SET-LOGON-PARAMETERS des Enter-Jobs angegeben ist, gestartet. Die letzte Aktion der kundenspezifischen Vorbereitungen muss das Kommando SHUTDOWN MODE=*NO/*END(RESTART=*NO) sein. Zur Ausführung des Kommandos SHUTDOWN wird das Privileg OPERATING benötigt.

Eine Muster-Kommandodatei für den Enter-Job wird als Release-Item
`SYSENT.SHUTDOWN.TEMPLATE` der Release-Unit BS2CP geliefert und mit IMON installiert. Die Systembetreuung muss diese Datei an die Systemumgebung und an die kundenspezifischen Beendigungsschritte anpassen und unter dem Namen `$TSOS.SYSENT.SHUTDOWN` bereitstellen.

Für weitere Informationen siehe Handbuch „Bedienen und Verwalten“ [59].

### 2.4.2 Außerplanmäßige Beendigung

Zur abnormaler Beendigung des Systemlaufs kann es z.B. aus folgenden Gründen kommen:

- Hardware-Fehler an peripheren Geräten oder am Server
- Software-Fehler durch Sättigungs- oder Deadlocksituationen
- Zustände, in denen das System selbst erkennt, dass eine sinnvolle Weiterarbeit nicht möglich ist

In diesen Fällen kann das System nicht ordnungsgemäß beendet werden, d.h. alle Jobs, die zu diesem Zeitpunkt tätig waren, müssen i.A. nach Wiederherstellung des Systems wiederholt werden.

In den meisten Fällen ist es notwendig, den Systemzustand zum Fehlerzeitpunkt mit dem Dienstprogramm SLED sicherzustellen, um eine spätere Diagnose zu ermöglichen.

Die Ausgabedateien für den SLED (siehe „Diagnosehandbuch“ [14]) dürfen auch außerhalb des Home-Pubsets liegen, jedoch nur auf Platten bzw. Pubsets, die als IPL-Platte bzw. Home-Pubset geeignet wären, also z.B. nicht auf SM-Pubsets oder DRV-Privatplatten.

Es ist möglich, dass es sich um eine Störung handelt, die nicht durch ein unmittelbares Neuladen des Systems zu beheben ist, z.B. wenn der Katalog oder andere wichtige Platteninhalte zerstört sind. In diesem Fall müssen die Platten wiederhergestellt werden (letzter Sicherungsstand, siehe Kapitel „Datensicherung“ auf Seite 531).

Belegungen von Privatplatten können nicht mehr zurückgegeben werden, d.h. das System bleibt im SVL der Platte eingetragen und muss ggf. mit dem Kommando UNLOCK-DISK ausgetragen werden.

Belegungen von Pubsets können ebenfalls nicht mehr aufgehoben werden. Bei der nächsten Systemeinleitung weist die Meldung `NSI424A` auf diese Situation hin. Mit der Antwort `ACCEPT` werden solche Pubsets wieder freigegeben.
Während der BS2000-Session kann der Operator mit dem Kommando UNLOCK-DISK noch existierende Dateilocks für einzelne Platten oder ganze Pubsets zurücksetzen.

### 2.4.3 Automatischer Restart

Die Systemfunktion „Automatischer Restart“ unterstützt den operatorlosen Betrieb. Sie ermöglicht, dass nach einem Systemabbruch das Betriebssystem automatisch neu geladen wird. Ein evtl. notwendiger Systemspeicherauszug mit SLED oder SNAP wird ohne Operatoreingriffe erstellt.

Der automatische Restart kann durch den Operator an- oder abgeschaltet werden (siehe Kommandos SET-RESTART-OPTIONS MODE=*ON/*OFF, DUMP=*SLED/*SNAP und SHOW-RESTART-OPTIONS). Standardmäßig ist er abgeschaltet.

#### Voraussetzungen für den automatischen Restart mit SLED

Die Voraussetzungen für einen FAST-Startup müssen erfüllt sein und es dürfen keine fehlerhaften Reps und Parameter vorliegen.

Bei einem Systemspeicherauszug mit SLED muss eine der beiden Dateien zur Verfügung stehen:

- $TSOS.SLEDFILE (bei SLED ohne Parameterdatei)
- $TSOS.SYSPAR.SLED.180 (bei SLED mit Parameterdatei)

Ist für den Fall des SLED ohne Parameterdatei in der Datei $TSOS.SLEDFILE bereits ein Speicherauszug enthalten (nicht logisch leer), kann kein neuer Speicherauszug erstellt werden und das System wird entsprechend den Vorgaben aus dem Kommando SET-RESTART-OPTIONS sofort neu geladen.

Die Dump-Datei $TSOS.SLEDFILE darf keine „große“ Datei ≥ 32 GB sein.

#### Voraussetzungen für den automatischen Restart mit SNAP

Die SNAP-Funktion muss eingeschaltet sein. Dies erfolgt über den Parameter SNAP-ACTIVE-SWITCH (Parametersatz SNAP) im Startup-Parameterservice (siehe Seite 118) oder im laufenden System durch das Kommando ACTIVATE-SNAPSHOT.

Wenn die Systemdatei $TSOS.SNAPFILE nicht vorhanden ist, dann wird sie beim Startup (Parameter SNAP-ACTIVE-SWITCH=ON) in Standardgröße bzw. bei Ausführung des Kommandos ACTIVATE-SNAPSHOT in der angegebenen Größe angelegt.

Informationen über den Status von SNAP liefert das Kommando SHOW-SNAPSHOT-STATUS.

# 3 Parameterservice

Über eine oder mehrere Startup-Parameterdateien werden verschiedene Software-Komponenten während des Startups mit Daten versorgt.

Es werden folgende Parameter eingelesen und an die betroffenen Routinen weitergeleitet. In Klammern sind jeweils die „Schlüsselwörter" innerhalb der Parameter-Anweisung zur Identifizierung der Software-Komponenten aufgeführt.

- Abrechnungssystem (ACCOUNT, Seite 82)
- Definition von ADAM-Geräten (ADAM, Seite 84)
- Inbetriebnahme der dynamischen Subsystemverwaltung (DSSM, Seite 85)
- Bestimmung der Systemzeit und der universellen Weltzeit (GTIME, Seite 87)
  Die Angabe der GTIME-Parameter ist notwendig. Ohne vorgegebene GTIME-Parameter wird bei jeder Systemeinleitung eine Frage an den Operator ausgegeben, die dieser beantworten muss.
- Änderung des Konfigurationszustandes (IOCONF, Seite 92)
- Voreinstellungen für NK-ISAM (ISAM, Seite 96)
- Festlegung des TSN-Modus (JMS, Seite 99)
- Verwaltung des Speichers (MEMORY, Seite 100)
- Festlegung der Bedienstationskonfiguration, Änderung der Bedienstationseigenschaften, Zuordnung von Berechtigungsschlüsseln und Filterstufen und Unterdrückung bestimmter Meldungen an Bedienstationen (OPR, Seite 104)
- Zuweisung des Seitenwechselbereichs bei Startup (PAGING, Seite 115)
- Snapshot-Initialisierung (SNAP, Seite 118)
- Voreinstellung von Systemparametern (SYSOPT-CLASS2, Seite 120)
- Änderung von Voreinstellungen der Systemeinleitung (SYSOPT-IPL, Seite 123)
- Festlegung der Parameter für das Datenkommunikationssystem (BCAM)
- Angabe des Sicherheitsbeauftragten bei Einsatz von SECOS (SRPM)
- Konfigurationsdaten für VM2000 (VM2000, S-Server)

Die Parametersätze werden im Folgenden in alphabetischer Reihenfolge beschrieben. Ausnahmen: BCAM und VM2000 werden in den jeweiligen Produkthandbüchern erläutert und SRPM im SECOS-Handbuch „Zugangs- und Zugriffskontrolle“ [48].

## 3.1 Auswahl der Parameterdatei

### Automatische Auswahl der Parameterdatei

Wenn ein System auf wechselnden Servern betrieben werden soll, kann je nach Server eine andere Parameterdatei verwendet werden.

Eine automatische Wahl dieser Parameterdatei erfolgt durch BS2000/OSD im **FAST**- und **AUTOMATIC**-Startup sowie im **DIALOG**-Startup bei Auswahl der Standard-Paramterdatei (Eingabe von `P.`) in folgender Reihenfolge:

- im Native-Betrieb und im VM2000-Monitorsystem:
  1. die Datei `$TSOS.SYSPAR.BS2.180.<name>` (für S-Server).
     `<name>` ist dabei der Wert des Operanden `NAME` aus der GEN-Anweisung von IOGEN, der in die ersten acht Zeichen des IOCF-Kommentarfeldes des aktiven IOCF eingetragen ist.
  2. die Datei `$TSOS.SYSPAR.BS2.180.<system-name>` (für SQ-Server).
     `<system-name>` ist dabei der Systemname, der in der BS2000-Konfiguration vergeben wurde; er kann im SVP-IPL-Menü nochmals verändert werden.
- in einem VM2000-Gastsystem:

  die Datei `$TSOS.SYSPAR.BS2.180.<vm-name>`.
  `<vm-name>` ist dabei der VM-Name des Gastsystems.
- wird keine spezifische Parameterdatei gefunden, wird nach der Datei mit dem Standardnamen `$TSOS.SYSPAR.BS2.180` gesucht.
- wird keine der o.g. Parameterdateien gefunden, so wird ein FAST- oder AUTOMATIC-Startup unterbrochen und es wird für die Dauer des Parameter-Einlesens auf DIALOG-Startup umgeschaltet.

### Dialoggesteuerte Auswahl der Parameterdatei

Beim **DIALOG-**Startup stehen dem Operator die Eingabequellen Platte und Bedienstation für die Zuweisung zur Verfügung. Die Zuweisung erfolgt mit Beantwortung der Meldung `NSI0050` (siehe Seite 66):

- Parametersätze können aus Dateien eingelesen oder direkt an der Konsole eingegeben werden.
- Hat die Systembetreuung die Parameterdatei auf einer Platte hinterlegt, dann erfolgt die Zuweisung über den entsprechenden Dateinamen.
- Durch die Antwort `P.CONS` auf die Meldung `NSI0050` wird auf die Eingabe von Bedienstation umgestellt. Folgende Meldung fordert den Operator zur Eingabe des Schlüsselwortes für die nachfolgenden Parametersätze auf:

  ```
  NSI4120 ENTER PARAM KEYWORD: (<KEYWORD>; ? (DISPLAY LIST OF POSSIBLE
          KEYWORDS); EOT (END))
  ```

  Der Operator hat die Möglichkeit, sich durch Angabe von „?“ alle möglichen Schlüsselwörter anzeigen zu lassen. Das Schlüsselwort muss direkt eingegeben werden. Die Ausgabe `BEGIN` wird intern für die Protokollierung in der CONSLOG-Datei generiert.

  Die Eingabe der einzelnen Parametersätze wird durch folgende Meldung angefordert:

  ```
  NSI4125 ENTER PARAM RECORD FOR KEYWORD (&00); (<PARAM>; /- (IGNORE LAST
          RECORD); EOT (END))
  ```

  Diese Meldung wird solange wiederholt, bis `EOT` eingegeben wird. In diesem Fall wird die Anweisung EOF intern generiert. Es folgt wiederum die Meldung `NSI4120`, die zur Eingabe des nächsten Schlüsselworts auffordert.

  Mit der Angabe von `EOT` wird die Eingabe von der Bedienstation beendet, ein „/END-PARAMS“-Satz wird ebenfalls intern generiert.

Das Einlesen der Datensätze erfolgt unmittelbar vor dem Laden des Klasse-1-Execs. Die Parameter- und Steuersätze werden komprimiert in einem Puffer im Startup-Lademodul gespeichert.

Alle Sätze, die über den Parameterservice verarbeitet wurden, werden in Form von Meldungen in der Protokolldatei CONSLOG erfasst.

Wird das Einlesen einer Parameterdatei mit Fehler abgebrochen, dann erhält der Operator an der Bedienstation nach einer Fehlermeldung die Meldung `NSI008F`. Der Operator kann angeben, ob die bereits eingelesenen Sätze erhalten bleiben oder ignoriert werden sollen. Danach erscheint die Meldung `NSI0050`. Ein Eingabegerät kann jeweils angegeben werden. Dieser Vorgang wiederholt sich solange, bis der Operator den Parameterservice mit der Eingabe `END` beendet. Stehen in der Parameterdatei fehlerhafte Anweisungen für den Parameterservice, so werden diese Anweisungen an der Bedienstation gemeldet und können sofort vom Operator korrigiert oder übergangen werden.

## 3.2 Aufbau und Inhalt einer Parameterdatei

Eine Parameterdatei setzt sich zusammen aus Parametersätzen, Kommentarsätzen und Steuersätzen.

Parametersätze sind Datensätze, die nur von der entsprechenden Software-Komponente interpretiert werden. Die Einleseroutine übernimmt die Parametersätze ungeprüft.
Kommentarsätze beginnen immer mit einem *(Stern), werden vom Parameterservice ignoriert und deshalb auch nicht protokolliert.
Steuersätze sind die Sätze, die Anweisungen für den Parameterservice enthalten.
Sie beginnen immer mit einem Schrägstrich.

Anweisung **BS2000 PARAMS**

Diese Anweisung ist immer die erste in der Parameterdatei (entfällt bei der Eingabe von der Bedienstation).

Anweisung **BEGIN <schlüsselwort>**

Das angegebene Schlüsselwort bestimmt die Zugehörigkeit der folgenden Parametersätze zur jeweiligen Software-Komponente. Mögliche Schlüsselwörter sind:

- ACCOUNT
- ADAM
- DSSM
- GTIME
- IOCONF
- ISAM
- JMS
- MEMORY
- OPR
- PAGING
- SNAP
- SYSOPT-CLASS2
- SYSOPT-IPL

Das Schlüsselwort SRPM wird vom Software-Produkt SECOS zur Festlegung der Benutzerkennung des Sicherheitsbeauftragten verarbeitet. Der Parametersatz wird im SECOS-Handbuch „Zugangs- und Zugriffskontrolle“ [48] beschrieben. Die Schlüsselwörter BCAM und VM2000 werden in den jeweiligen Produkthandbüchern erläutert.

Anweisung **EOF**

Diese Anweisung schließt den Parameterabschnitt für die mit <schlüsselwort> angesprochene Software-Komponente ab.

Anweisung **ADD**

Eine Parameterdatei kann auch die Namen weiterer Parameterdateien enthalten, die verwendet werden sollen. Die mit den ADD-Anweisungen angegebenen Dateien werden eingelesen, sobald die Parameterdatei, die die ADD-Anweisungen enthält, vollständig eingelesen ist.

Es dürfen maximal 16 ADD-Anweisungen angegeben werden.
Nach einer ADD-Anweisung sind keine Anweisungen außer der ADD-Anweisung selber und der END-PARAMS-Anweisung erlaubt. In einer über ADD definierten Parameterdatei können keine weiteren ADD-Anweisungen angegeben werden.

Anweisung **END-PARAMS**

Diese Anweisung ist die letzte in der Parameterdatei.

Die zu einer Software-Komponente gehörenden Parametersätze müssen nicht zwingend alle hintereinander angegeben sein. Es dürfen mehrere BEGIN-Anweisungen mit demselben Schlüsselwort und darauffolgenden Parametersätzen und EOF-Anweisungen in der Parameterdatei stehen.

Als Muster der Systemparameterdatei wird die Datei SYSPAR.BS2.180.TEMPLATE ausgeliefert. Sie enthält

- die Anweisungen BEGIN <schlüsselwort> und EOF
- Mustervorgaben für Parameter, wenn ohne Vorgaben keine Standardwerte gültig werden (betrifft nur den Parametersatz für GTIME)

  Nur für GTIME sind explizite Angaben notwendig. In die Musterdatei wurden die zum Zeitpunkt der Versionsfreigabe bekannten gesetzlichen Zeitangaben für drei Zeitzonen aufgenommen (Zeitzonen 0, 1 und 2, einschl. der Informationen zur Sommer-/Winterzeit-Umstellung).
  Bei Betrieb von BS2000/OSD in einer dieser Zeitzonen können die Angaben für die jeweils anderen Zeitzonen gelöscht werden (bei Betrieb in der Zeitzone 3 müssen die Kommentarzeichen entfernt werden).
  Bei Betrieb in einer anderen als den angegebenen Zeitzonen müssen die Angaben in gleicher Form nach den gültigen Umstellzeiten vom Kunden selber erstellt werden.

Die Standardwerte für den Parametersatz OPR haben zur Folge, dass alle Konsolmeldungen, die eine explizite Routingvorgabe brauchen, nicht auf der Hauptkonsole ausgegeben werden.

Die Musterdatei muss zum Betrieb als Systemparameterdatei zu SYSPAR.BS2.180 umbenannt werden. Sie kann um kundenspezifische Parametervorgaben erweitert werden.

**Beispiel** für eine Parameterdatei

```
/BS2000 PARAMS

*Accouning system
/BEGIN ACCOUNT
/START-ACC NAME=$RZ.
/EOF

*Paging
/BEGIN PAGING
PAGING VOLUME=(PUBA04,PUBA05)
/EOF

*Memory management
/BEGIN MEMORY
SHRSIZE 6,UNIT=1MB
ASAMCTRL 4
/EOF

*System time and season
/BEGIN GTIME
ZONE=+01:00
DIFF=1:00
SEASON=S
EPOCH=00
CHDATE=1900-01-01/00:00
CHDATE=2011-03-27/02:00
CHDATE=2011-10-30/03:00
CHDATE=2012-03-25/02:00
CHDATE=2012-10-28/03:00
/EOF

*TSN mode
/BEGIN JMS
TSN-MODE=A
/EOF

*DSSM
/ADD PAR.FILE.DSSM

/END-PARAMS
```

Jede auf Platte befindliche Parameterdatei muss eine SAM-Datei mit variablen Sätzen in Standard-Blockung sein. Die Satzlänge muss ohne Satzlängenfeld 1-72 Byte sein, mit Satzlängenfeld demnach 5-76 Byte.
Für BLKCTRL=PAMKEY gilt, dass in den ersten 2 Byte eines Blockes die sedezimale Blocklänge, in den ersten 2 Byte eines Satzes die sedezimale Satzlänge (d.h. Satzformat V) erwartet wird.
Im NONKEY-Modus wird die Blocklänge aus dem simulierten Schlüssel innerhalb des Datenfeldes geholt (Länge 4 Byte).

Parameter-Eingabedateien können z.B. mit EDT erstellt, fortgeschrieben und gemischt werden.

## 3.3 Starten des Abrechnungssystems (ACCOUNT)

Das Abrechnungssystem kann im Rahmen der Systemeinleitung oder später per Kommando START-ACCOUNTING aktiviert werden, indem die dazu notwendigen Anweisungen über den Parameterservice mitgeteilt werden. Die Systembetreuung muss für diesen Zweck in der Parameterdatei mit dem standardmäßigen Namen SYSPAR.BS2.180[.name] die entsprechenden Angaben bereitstellen.

Wird das Abrechnungssystem über den Parameterservice aktiviert und liegt die Abrechnungsdatei auf einem anderen als dem Home-Pubset, kann das zu folgendem Problem führen: Die Datei kann nicht eröffnet werden, weil das Importieren dieses Pubsets noch nicht abgeschlossen ist. Deshalb muss in diesem Fall die Abrechnungsdatei generell auf dem Home-Pubset liegen.

Schlüsselwort für den BEGIN-Satz ist **ACCOUNT**.
Die Maximalzahl der dafür zugelassenen Parametersätze beträgt 16.

Der Parametersatz beginnt wahlweise mit der Anweisung START-ACCOUNTING bzw. STOP-ACCOUNTING. Die Anweisungen und die dazugehörigen Operanden können bis zur Eindeutigkeit abgekürzt werden.

Anweisung **START-ACCOUNTING**

Die Anweisung /START-ACCOUNTING aktiviert das Abrechnungsverfahren, wobei grundsätzlich die selben Operanden vereinbart werden können wie beim Kommando START-ACCOUNTING.

**Format des Parametersatzes zum Starten des Abrechnungssystems**

| Anweisung | Operanden |
|---|---|
| /START-ACC[OUNTING] | [NAME = <u>*STD</u> / file] [,SPACE = <u>*STD</u> / (primary,secondary)]<br>[,BLKSIZE = <u>*STD</u> / (*STD,n)] [,VOLUME = <u>*STD</u> / vsn]<br>[,ALT[ERNATIVE-FILES] = <u>*NONE</u> / file / (file,...)]<br>[,SET[-RECORD-TYPE] = <u>*STD</u> / *ALL / record / (record,...)]<br>[,ADD[-RECORD-TYPE] = <u>*NONE</u> / record / (record,...)]<br>[,REMOVE[-RECORD-TYPE] = <u>*NONE</u> / record / (record,...)]<br>[,ACCOUNTING-PERIOD = <u>*STD</u> / period]<br>[,JOB-CLASS = <u>*NONE</u> / *ALL / job-class / (job-class,...)] |

Die Anweisung START-ACCOUNTING kann mit dem Fortsetzungszeichen Bindestrich auf mehrere Zeilen aufgeteilt werden.

Die Fortsetzungszeilen müssen syntaktisch abgeschlossen sein, da sie vom System einzeln überprüft werden. Deshalb dürfen bei der Anweisung START-ACCOUNTING die Operanden mehrfach angegeben werden.

Da es vorkommen kann, dass nicht alle Werte einer Liste in eine Zeile passen, wirken die Operanden ADD-RECORD-TYPE, REMOVE-RECORD-TYPE und ALTERNATE-FILES ergänzend. Die Liste kann dann in der nächsten Zeile mit gleichen Operanden fortgesetzt werden (der Operand SET-RECORD-TYPE wird mit dem Operanden ADD-RECORD-TYPE fortgesetzt).

Anweisung **STOP-ACCOUNTING**

Die Anweisung /STOP-ACCOUNTING vereinbart, dass im Systemlauf keine Abrechnungssätze gesammelt werden sollen.

**Ausschnitt aus der Parameterdatei**

```
/BS2000 PARAMS
:
/BEGIN ACCOUNT
/START-ACCOUNTING NAME=$RZ.,- ———————————————————————————————————— (1)
SPACE=(99,99),- ————————————————————————————————————————————————— (2)
SET=*ALL,JOB-CLASS=*ALL ——————————————————————————————————————— (3)
/EOF
:
/END-PARAMS
```

(1) Die Abrechnungsdatei wird unter der Benutzerkennung RZ mit ihrem Standardnamen SYS.ACCOUNT.<date>.xxx.nn katalogisiert, wobei <date> in der Form yy.mm.dd oder yyyy-mm-dd auftreten kann, abhängig von der Einstellung des Systemparameters FMTYFNLG.

(2) Die Speicherplatzzuweisung beträgt primär und sekundär 99 PAM-Blöcke.

(3) Sämtliche Abrechnungssätze werden geschrieben, und die Daten aller Jobklassen sollen periodisch in der Abrechnungsdatei erfasst werden.

Zum Abrechnungssystem von BS2000/OSD siehe auch das Kapitel „Accounting“ auf Seite 561.

## 3.4 Parametersatz ADAM

Die Informationen der ADT-Anweisungen (IOGEN) müssen auch im Parameterservice durch die Systembetreuung festgelegt werden.



Die benötigten ADT-Anweisungen müssen sowohl bei der Hardware-Generierung mit IOGEN, als auch in den Parametersätzen **vorhanden sein und übereinstimmen**. Eine automatische Überprüfung findet nicht statt.

Das Schlüsselwort für die Definition von ADAM-Geräten ist **ADAM**.
Die Maximalzahl der dafür zugelassenen Parametersätze beträgt 16.
Fehlerhafte Parametersätze können im Dialog im Startup korrigiert werden.

### Anweisungsformat

| Anweisung | Operanden | Bedeutung |
|---|---|---|
| ADT | (type,name) | Gerätetypcode und symbolischer Name des ADAM-Geräts |

Es darf im Gegensatz zur gleichlautenden IOGEN-Anweisung pro Anweisung nur ein Paar (type,name) angegeben werden. Wird die ADT-Anweisung für ein ADAM-Gerät mehrfach angegeben, so wird der zuletzt korrekt eingegebene symbolische Name verwendet.

**type**
Gerätetypcode des ADAM-Geräts. Zulässiger Wertebereich: 71, 72, ..., 7F.

**name**
Symbolischer Name des ADAM-Geräts.
Wertebereich: Die dem BS2000/OSD bekannten ADAM-Gerätenamen oder die mittels der IOGEN-Anweisung EVA bei der Hardware-Generierung neu definierten ADAM-Gerätenamen.

## 3.5 Inbetriebnahme der dynamischen Subsystemverwaltung (DSSM)

Die dynamische Subsystemverwaltung wird während der BS2000-Systemeinleitung gestartet.
Über den Parameterservice werden alle zur Initialisierung von DSSM notwendigen Informationen mitgegeben. Dazu zählen der Name des statischen Subsystemkataloges und die DSSM-Versionsnummer. Des Weiteren kann bereits – wenn unbedingt erforderlich – die Protokollierung von DSSM-spezifischen Daten für die Fehlerdiagnose eingeschaltet werden.

Schlüsselwort für die Inbetriebnahme der Subsystemverwaltung ist **DSSM**.
Die Maximalzahl der dafür zugelassenen Parametersätze beträgt 16.

#### Format des Parametersatzes zur Inbetriebnahme der dynamischen Subsystemverwaltung

| **Format** | **Bedeutung** |
|---|---|
| SSMCAT = name | Name des statischen Subsystemkatalogs |
| VERSION = version | Versionsnummer von DSSM |
| LOGGING = ON / <u>OFF</u> | Steuert die DSSM-spezifische Protokollierung zur Fehlerdiagnose |

Subsysteme, die nicht während der Systemeinleitung automatisch aufgebaut werden, können im BS2000-Systemlauf mit dem Kommando START-SUBSYSTEM gestartet werden.

### Ausschnitt aus der Parameterdatei

```
/BS2000 PARAMS
:
/BEGIN DSSM
SSMCAT=<name> ————————————————————————————————————— (1)
VERSION=<version> ————————————————————————————————— (2)
LOGGING=OFF ——————————————————————————————————————— (3)
/EOF
:
/END-PARAMS
```

(1) Jede DSSM-Version kann Subsystemkataloge verwenden, die mit bestimmten SSCM-Versionen generiert wurden:

| | SSCM | | | | |
|---|---|---|---|---|---|
| DSSM | **V 1.0** | **V2.0** | **V2.1** | **V2.2** | **V 2.3** |
| **V 3.0** | x | | | | |
| **V 3.5** | x | x | | | |
| **V 3.6** | x | x | x | | |
| **V 3.8** | x | x | x | x | |
| **V3.9 / V4.0 / V4.1 / V4.3** | x | x | x | x | x |

Die Steuer- und Parametersätze müssen in der Parameterdatei nur vorhanden sein, wenn die Systembetreuung von folgenden Standardwerten abweichen will: SSMCAT=$TSOS.SYS.SSD.CAT.X, VERSION=043 und LOGGING=OFF.

(2) Die Versionsnummer bezieht sich auf alle DSSM-spezifischen Dateinamen (z.B. SYSLNK.DSSM.043, SYSREP.DSSM.043) und wird dreistellig angegeben (z.B. 043).

(3) Die Anweisung LOGGING=OFF (Standardwert) bewirkt, dass die Protokollierung ausgeschaltet ist; (bei LOGGING=ON würde bereits während der Inbetriebnahme von DSSM ein Protokoll der Diagnosedaten erzeugt).

## 3.6 Steuerung der Systemzeit (GTIME)

In der Parameterdatei werden die Initialisierungsdaten für das Subsystem GET-TIME hinterlegt, das dem Benutzer über die Systemfunktion GTIME Informationen zur standardisierten Weltzeit und der Zeitverschiebung liefert. Auch das Betriebssystem benötigt diese Information.

Zur Initialisierung und Verwaltung der Systemzeit siehe das Kapitel „Systemzeit-Verwaltung“ auf Seite 705.

Die Systembetreuung legt über verschiedene Parameter die Relation zwischen der Systemzeit (lokale Zeit) und der universellen Weltzeit UTC fest. Damit kann das System und der Benutzer des Subsystems GET-TIME sowohl auf ein lokales als auch auf ein systemübergreifend verständliches Zeitbezugssystem (UTC) zugreifen.
Ohne diese Daten (aus der Parameterdatei oder im Dialog über Bedienstation) kann die Systemeinleitung **nicht** durchgeführt werden.

Die SVP-Uhr muss, außer bei automatischem Restart oder bei Gastsystembetrieb, zur Systemeinleitung die korrekte lokale Zeit (Systemzeit) enthalten.

Schlüsselwort für die Festlegung der Relation zwischen der Systemzeit und der universellen Weltzeit in der Parameterdatei ist **GTIME**.
Die maximale Anzahl der dafür zugelassenen Parametersätze beträgt 256.

**Format des Parametersatzes zur Steuerung der Systemzeit**

| Format | Bedeutung |
|---|---|
| [NEXTZONE] | Es beginnt ein neuer GTIME-Parameterblock |
| ZONE = +hh:mm / -hh:mm | Zeitzone |
| DIFF = h:mm | Größe des Zeitsprungs (Umstellungsdifferenz) |
| SEASON = S / W | Angabe Sommer-/Winterzeit vor dem ersten Umstellungszeitpunkt |
| EPOCH = 00 / nn | Angabe der Epoche für das TODR (2 Sedezimalzeichen). EPOCH=00 bezeichnet die Standard-Epoche 1.1.1900 - 17.9.2042. |
| CHDATE = yyyy-mm-dd/hh:mm<br>:<br>CHDATE = yyyy-mm-dd/hh:mm | Umstellungszeitpunkt 1<br>:<br>Umstellungszeitpunkt n (max. 125) |



Umstellungszeitpunkte können auch im laufenden BS2000-Betrieb mit den Kommandos ADD-/MODIFY-/REMOVE-/SHOW-CHANGE-DATE verwaltet werden. Änderungen von Umstellungszeitpunkten durch die Kommandos müssen aber manuell in die Parameterdatei eingetragen werden, wenn sie im nächsten Systemlauf gültig sein sollen.

**NEXTZONE**
Trennt die GTIME-Parameter verschiedener Zeitzonen voneinander ab. Dadurch können die Daten für mehrere Zeitzonen in der GTIME-Parameterdatei enthalten sein.
Die Angabe dieses Operanden kann entfallen, wenn in der Parameterdatei nur die Daten für eine Zeitzone enthalten sind.

**ZONE = -/+hh:mm**
Zeitzone in Stunden und Minuten.
Dieser Wert beschreibt die lokale, gesetzliche Zeitzone im Vergleich zur „Greenwich Mean Time“, der UTC (Universal Time Coordinate).
Wertebereich: -12:00 ≤ hh:mm ≤ +11:59
Beispielsweise beträgt der Vorsprung der mitteleuropäischen Zeit gegenüber der UTC eine Stunde; als Wert ist folglich +01:00 anzugeben.
Der Wert ZONE **muss** in der Parameterdatei angegeben werden.

**DIFF = h:mm**
Größe des Zeitsprungs bei einer Umstellung zwischen Sommer- und Winterzeit.
Wertebereich: 0:00 ≤ h:mm ≤ 9:59
Der Wert DIFF **muss** in der Parameterdatei angegeben werden.
Ist DIFF nicht 0:00, müssen die Operanden SEASON und – mindestens ein – CHDATE angegeben werden.

**SEASON = S / W**
Gibt an, ob vor dem Zeitpunkt der **ersten** Umstellung **S**ommer- oder **W**interzeit gültig war. (Als „Winterzeit“ wird hier die eigentliche Normalzeit bezeichnet: die von der Normalzeit abweichende „Daylight-Saving-Time“ wird „Sommerzeit“ genannt.)
Dieser Wert muss für die interne Zeitberechnung angegeben werden, wenn mit dem Operanden CHDATE eine Umstellung der Zeit vereinbart wird. Von diesem Ausgangswert muss die Systemfunktion GTIME, auch nach mehreren Umstellungen, auf die jeweils gültige Zeit schließen.

Dieser Wert wird von der Systemfunktion CTIME nicht ausgewertet. Da andere Systemfunktionen (z.B. JMS, DVS) intern CTIME benutzen, sind die beim Operanden CHDATE aufgeführten Hinweise zu CTIME und SEASON zu berücksichtigen.

**EPOCH = <u>00</u>** / xx
Gibt die Epoche für das TODR an (2 Sedezimalzeichen).
EPOCH=00 bezeichnet die Standard-Epoche 1.1.1900 - 17.9.2042, siehe Seite 709.



Voraussetzung für den Einsatz einer neuen TODR-Epoche ist es, dass die dadurch nicht mehr verfügbaren alten TODR-Zeitstempel, die vor dem Beginn der aktuellen Epoche lagen (und jetzt anders interpretiert würden), nicht mehr gebraucht werden. Dies zu beurteilen, ist Aufgabe der Systemverwaltung.
„Alte“ Zeitstempel können (bei Kenntnis der jeweils eingestellten Epochen) aber nach Umrechnung mit „neuen“ Zeitstempeln verglichen werden, siehe Abschnitt „Rechnen mit TODR-Epochen“ auf Seite 711.

Zeitstempel im TODX-Format können stets verglichen werden. TODX-Werte sind im Jahres-Intervall (1900...4317) monoton steigend.

**CHDATE = yyyy-mm-dd/hh:mm**
Vereinbarung von Umstellungszeitpunkten (1..125). Das erste Datum muss mit 1900 beginnen, die nachfolgenden müssen lückenlos und zeitlich aufsteigend sein (siehe Beispiel auf der nächsten Seite).
Format und Wertebereich des Datums:

| | | |
|---|---|---|
| yyyy | : Jahr | 1900 ≤ yyyy < 2042 |
| mm | : Monat | 1 ≤ mm ≤ 12 |
| dd | : Tag des Monats | 1 ≤ dd ≤ 31 |
| hh | : Stunde | 0 ≤ hh ≤ 23 |
| mm | : Minute | 0 ≤ mm ≤ 59 |

Die Datumsangaben dienen dem System beim Startup zur Bestimmung, ob momentan Sommer- oder Winterzeit vorherrscht, und somit zur Festlegung der Differenz zwischen lokaler Zeit und der Weltzeit UTC. Damit kann die Parameterdatei über mehrere Umstellungszeitpunkte hinweg verwendet werden.

Weiterhin werden die Umstellungszeitpunkte für die CTIME-Funktion benötigt, die u.a. Zeitangaben von lokaler Zeit in UTC umwandelt. CTIME muss bei der Interpretation von lokalen Zeitstempeln davon ausgehen, dass von 1900-01-01/00:00 bis zum ersten CHDATE Winterzeit herrscht, auch wenn z.B. das erste CHDATE 1994-09-25/03:00 lautet und SEASON=S angegeben wurde. Es werden dann aus Benutzersicht Zeitstempel vor dem ersten CHDATE fälschlicherweise als Winterzeitstempel interpretiert.
Um diese Zweifelsfälle zu reduzieren, ist es sinnvoll, die Liste bzgl. der zurückliegenden Umstellungszeitpunkte möglichst vollständig zu halten. Optimal für die einwandfreie Funktion der CTIME-Schnittstelle ist es, als erstes Datum 1900-01-01/00:00 mit SEASON=S einzutragen. Das zweite CHDATE muss dann eine Winter- nach Sommerzeitumstellung festlegen. Dann kann für alle zurückliegenden Zeitangaben entschieden werden, ob sie als Sommer- oder Winterzeit interpretiert werden müssen.

Die Differenz zwischen zwei Umstellungszeitpunkten muss im Bereich von 4 bis 8 Monaten liegen (Ausnahme: die Differenz zwischen dem CHDATE 1900-01-01/00:00 und dem zweiten CHDATE darf beliebig groß sein).



Unrichtige GTIME-Parameter verfälschen die Zeitrelation von Systemzeit zur universellen Weltzeit UTC und haben ebenso fatale Auswirkungen wie eine falsch eingestellte SVP-Uhr!

Die Zeitumstellung erfolgt unterbrechungsfrei, d.h. das System wird kontinuierlich über eine Zeitumstellung hinweg betrieben, weil das TODR nicht mehr die genaue lokale Zeit enthalten muss. Die lokale Zeit wird aus dem Inhalt des TODR und einem Korrekturwert ermittelt (siehe Kapitel „Systemzeit-Verwaltung“ auf Seite 705).

Die zentral ausgelieferte Parameterdatei wird mit den GTIME-Parametern für mehrere Zonen verarbeitet. Um die Parameter für die richtige Zeitzone auszuwählen, sollte an S-Servern die Differenz zwischen UTC zur vorliegenden Zeitzone am SVP eingestellt werden. Auf SQ-Servern ist die Zeitzone im Trägersystem eingestellt. Die Zeitzone wird auf allen Servern aus der SVP-Zeit (STORE REAL CLOCK) ermittelt (siehe Kapitel „Systemzeit-Verwaltung“ auf Seite 705).

Wenn die Zeitzone nicht aus der SVP-Zeit ermittelt werden kann (Fehlerfall), dann wird der Eintrag im SVL der Startup-Platte zur Auswahl der richtigen Parameter herangezogen. Wenn auch dort kein gültiger Eintrag vorhanden ist und mehr als eine Zeitzone in den Parametersätzen enthalten ist, dann wird der Operator mit den Meldungen `ETMGT30` und `ETMGT31` nach einer Zonenangabe gefragt. Alternativ kann er die Systemeinleitung beenden.
Wenn die Zeitzone weder durch die SVP-Zeit noch durch das SVL der Startup-Platte bestimmt werden kann, die Parameterdatei aber genau eine Zeitzone enthält, dann wird diese als einzustellende Zeitzone übernommen.

Wenn eine Zeitzone durch die SVP-Zeit oder das SVL bestimmt ist, es aber keinen passenden Parametersatz dazu gibt, dann wird der Operator mit den Meldungen `ETMGT35` und `ETMGT36` nach der einzustellenden Zeitzone gefragt. Alternativ kann er die Systemeinleitung beenden.

In allen Fällen, in denen die Systemeinleitung fortgesetzt wird, wird die erkannte bzw. akzeptierte Zeitzone im SVL der Startup-Platte hinterlegt.

### Ausschnitt aus der Parameterdatei

```
/BS2000 PARAMS
:
/BEGIN GTIME
ZONE=+01:00 ──────────────────────────────────────── (1)
DIFF=1:00 ────────────────────────────────────────── (2)
SEASON=S ─────────────────────────────────────────── (3)
EPOCH=00 ─────────────────────────────────────────── (4)
CHDATE=1900-01-01/00:00 ──────────────────────────── (5)
CHDATE=1980-04-06/02:00
CHDATE=1980-09-28/03:00
CHDATE=1981-03-28/02:00
:
CHDATE=2009-03-22/02:00───────────────────────────── (6)
CHDATE=2009-10-25/03:00
CHDATE=2010-03-28/02:00
CHDATE=2010-10-31/03:00
CHDATE=2011-03-27/02:00
CHDATE=2011-10-30/03:00
CHDATE=2012-03-25/02:00
CHDATE=2012-10-28/03:00
:
/EOF
:
/END-PARAMS
```

(1) Es ist die mitteleuropäische Zeitzone angegeben.

(2) Der Differenzwert von einer Stunde bezeichnet die Höhe des Sprungs, der bei einer Umstellung zwischen Sommer- und Winterzeit erfolgt.

(3) Vor dem ersten CHDATE muss zwingend Sommerzeit eingestellt sein.

(4) Es gilt die Standard-Epoche 1.1.1900 - 17.9.2042 für das TODR.

(5) Pseudo-CHDATE: Er bewirkt, dass bis zum ersten wirklichen CHDATE Winterzeit gilt. Das entspricht der CTIME-Philosophie, die davon ausgeht, dass von 1900 an bis zum ersten eingetragenen CHDATE Winterzeit herrscht.
In diesem Beispiel (Nordhalbkugel) ist die Winterzeit Normalzeit.

(6) Zukünftige Umschalttage werden eingetragen.

## 3.7 Parametersatz IOCONF

Die Ein-/Ausgabe-Tabellen von BS2000/OSD-BC werden bei Startup aufgebaut.
Die Kanalperipherie wird aus dem aktiven IOCF ermittelt. Die Bus- und FC-Peripherie bei SQ-Servern wird über X2000 bestimmt.

Reihenfolge der Ermittlung des Konfigurationszustandes bei Startup:

1. Aus den Voreinstellungen für die Busperipherie (nur SQ-Server):
   ATTACHED: Bedienstationen, DFÜ-Geräte, Geräte der Familie '50' und alle Plattengeräte; Um Platten standardmäßig DETACHED zu generieren, kann ein optionaler Rep eingesetzt werden.
   DETACHED: alle übrigen Geräte.
2. Die Konfigurationszustände der Geräte werden aus den IOGEN-Daten im IOCF übernommen. Die Voreinstellungen für die Kanäle, Steuerungen und alle Verbindungen sind:
   ATTACHED: Kanäle und Mehrfachsteuerungen (Controller)
   INCLUDED: alle Verbindungen zwischen den Hardware-Einheiten
3. Aus dem Startup-Parameterservice (IOCONF) über die Anweisungen:
   MOD-IO-UNIT Konfigurationszustand für eine Hardware-Einheit festlegen
   MOD-IO-PATH Konfigurationszustand für eine Verbindung festlegen
4. Durch automatische Rekonfiguration im Startup:
   ATTACHED: alle Platten des Home-Pubsets, Paging-Platten, IPL-Konsole
   DETACHED: alle Bandgeräte
   Die impliziten Zustände aller Hardware-Einheiten und Verbindungen werden ermittelt. Bei TRANSDATA-Geräten darf nur ein Pfad im Zustand INCLUDED sein.

Das Schlüsselwort für die Änderung von Konfigurationszuständen ist **IOCONF**.
Die Maximalzahl der dafür zugelassenen Parametersätze beträgt 256.

Fehlerhafte Parametersätze werden ignoriert. Nach Beendigung des Startup wird die Meldung `NDI0550` ausgegeben, die die Anzahl der empfangenen Parametersätze und die Anzahl der fehlerhaften Sätze enthält. Bis zu 10 fehlerhafte Sätze werden mit der Meldung `NDI0551` ausgegeben, zusammen mit der Information, welche Art von Fehler vorliegt.

Die über den Parameterservice eingestellten Werte gelten für die laufende Session.

| Anweisung | Bedeutung |
|---|---|
| MOD-IO-UNIT | Konfigurationszustand einer Hardware-Einheit ändern |
| MOD-IO-PATH | Konfigurationszustand einer Verbindung zwischen Hardware-Einheiten ändern |

Tabelle 4: Übersicht der Anweisungen zum Parametersatz IOCONF

## 3.7.1 Anweisung MOD-IO-UNIT

### Format zur Änderung des Konfigurationszustandes einer Hardware-Einheit

| Anweisung | Operanden | Bedeutung |
|---|---|---|
| MOD-IO-UNIT | *class(unit) | Klasse und mnemotechnischer Gerätename der Hardware-Einheit |
| | ,STATE = ATT / DET / INV | Neuer Konfigurationszustand der Hardware-Einheit |
| MOD-IO-UNIT | *DEV(*RA[NGE](unit,range)) | Bereich von Hardware-Einheiten |
| | ,STATE = ATT / DET / INV | Neuer Konfigurationszustand der Hardware-Einheiten des Bereichs |



Mit Ausnahme der explizit angegebenen Abkürzungsmöglichkeiten müssen alle anderen Angaben vollständig ausgeschrieben werden. Auch das Weglassen von Operandennamen (Stellungsschreibweise) ist nicht zulässig.

***class(unit)**
Klasse und mnemotechnischer Gerätename der Hardware-Einheit, für die der Konfigurationszustand angegeben wird.

Für die Klasse *class können folgende Werte angegeben werden:

- `*CHA` bzw. `*CHN` (Kanal, Channel)
- `*CON` bzw. `*CTL` (Mehrfachsteuerung, Controller)
- `*DEV` bzw. `*DVC` (Endgerät, Device)

unit (2 oder 4 Zeichen) ist der mnemotechnische Gerätename einer Hardware-Einheit der angegebenen Klasse.

***DEV(*RA[NGE](unit,range))**
Mnemotechnischer Gerätename einer bestimmten Hardware-Einheit und Anzahl der darauf folgenden Geräte, für die der Konfigurationszustand angegeben wird.

unit (2 oder 4 Zeichen) ist der mnemotechnische Gerätename einer Hardware-Einheit.

range ist ein dezimaler Wert zwischen 2 und 999, der die Anzahl der Geräte angibt, für die – angefangen bei unit – der Konfigurationszustand geändert werden soll.

Die range-Angabe ist nur für Endgeräte (*class = *DEV/*DVC) erlaubt.

Bei 4-Byte-Mnemoniks wird ab unit hexadezimal weitergezählt, bei 2-Byte-Mnemoniks wird ab unit alphabetisch und anschließend mit den Ziffern 0 .. 9 weitergezählt.

range = 1 wird ignoriert, range = 0 mit Syntaxfehler abgewiesen.

Voraussetzung ist die Existenz der ersten Mnemonik (angegeben bei unit) im System.

Alle folgenden Mnemoniks werden ignoriert, wenn sie nicht im System vorhanden sind.

*Beispiele*

| | |
|---|---|
| *RANGE(A100,32) | A100, A101, ..., A11F |
| *RA(A1FF,5) | A1FF, A200, A201, A202, A203 |
| *RA(FA,40) | FA, FB, ..., FZ, F0, F1, F2, ..., F9, GA, GB, GC, GD |

**STATE =**
Bestimmt den neuen Konfigurationszustand der Hardware-Einheit.

**STATE = ATT / DET / INV**
Die Hardware-Einheit ist im Zustand ATTACHED (zugeschaltet), DETACHED (weggeschaltet) oder INVALID (ungültig).

### 3.7.2 Anweisung MOD-IO-PATH

**Format zur Änderung des Konfigurationzustandes von Verbindungen**

| Anweisung | Operanden | Bedeutung |
|---|---|---|
| MOD-IO-PATH | FROM = *class(unit) | Ausgangspunkt der Verbindung |
| | ,TO = *class(unit) | Endpunkt der Verbindung |
| | ,STATE = INC / REM / INV | Neuer Konfigurationszustand der Verbindung |



Mit Ausnahme der explizit angegebenen Abkürzungsmöglichkeiten müssen alle anderen Angaben vollständig ausgeschrieben werden. Auch das Weglassen von Operandennamen (Stellungsschreibweise) ist nicht zulässig.

**FROM = *class(unit)**
Klasse und mnemotechnischer Gerätename des Ausgangspunkts der Verbindung, für die der Konfigurationszustand angegeben wird.
Für die Klasse *class können folgende Werte angegeben werden:

- `*CHA` bzw. `*CHN` (Kanal, Channel)
- `*CON` bzw. `*CTL` (Mehrfachsteuerung, Controller)
- `*DEV` bzw. `*DVC` (Endgerät, Device)

unit (2 oder 4 Zeichen) ist der mnemotechnische Gerätename einer Hardware-Einheit der angegebenen Klasse.

**TO = *class(unit)**
Klasse und mnemotechnischer Gerätename des Endpunkts der Verbindung, für die der Konfigurationszustand angegeben wird.
*class(unit) siehe FROM.

**STATE =**
Bestimmt den neuen Konfigurationszustand der Verbindung.

**STATE = INC / REM / INV**
Die Verbindung ist im Zustand INCLUDED (zugeschaltet), REMOVED (weggeschaltet) oder INVALID (ungültig).

## 3.8 Voreinstellungen für NK-ISAM (ISAM)

Die indexsequenzielle Zugriffsmethode NK-ISAM des DVS unterstützt die Verwendung von PAM-Key-losen Platten. Darüber hinaus bietet NK-ISAM Möglichkeiten zur Performance-Optimierung durch die Einsparung von Platten-Ein-/Ausgaben. Dazu müssen die Pufferbereiche von NK-ISAM, die ISAM-Pools, geeignet dimensioniert werden.

Diese **ISAM-Pools** werden im privilegierten Adressraum oder in Data-Spaces angelegt und dienen als Zwischenspeicher für die Verarbeitung einer oder mehrerer NK-ISAM-Dateien. ISAM-Pools können sowohl vom Benutzer explizit über spezielle Makros bzw. Kommandos angelegt und verwaltet werden als auch implizit vom System. Sowohl der Benutzer als auch das System können taskspezifische oder taskübergreifende (Benutzerkennungs-, Benutzergruppen- oder Server-spezifische) ISAM-Pools anlegen.
Nicht-taskspezifische ISAM-Pools werden automatisch dateispezifisch in einem Data Space angelegt.

Taskspezifische ISAM-Pools können nur von der aufrufenden Task genutzt werden, Server-spezifische ISAM-Pools definieren einen Pufferbereich, der über Angabe des Pool-Namens auch anderen Jobs zugänglich ist.

Wird eine ISAM-Datei eröffnet, ohne dass sie vom Benutzer im Kommando oder Makro einem bestimmten ISAM-Pool zugeordnet wurde, verwendet das DVS für die Verarbeitung einen Standard-ISAM-POOL des Systems:
Bei einem OPEN mit SHARUPD=NO wird ein taskspezifischer Standard-ISAM-POOL zugeordnet, bei einem OPEN mit SHARUPD=YES ein taskübergreifender Standard-ISAM-POOL.

Die Größe eines Standard-ISAM-Pools legt die Systembetreuung in der Parameterdatei fest. Dort wird ebenso die Größe eines benutzerdefinierten ISAM-Pools festgelegt, wenn beim Anlegen des ISAM-Pools keine Angaben zu dessen Größe gemacht wurden.

Schlüsselwort für den BEGIN-Satz ist **ISAM**.
Die Maximalzahl der dafür zugelassenen Parametersätze beträgt 16.

Die Parameter für NK-ISAM können mehrfach angegeben werden. Gültig ist jeweils der zuletzt angegebene Wert.

### Format des Parametersatzes zur Voreinstellungen für NK-ISAM

| Parameter | |
|---|---|
| LCLDFPS = zahl | |
| GLBDFPS = zahl | *wird nicht mehr ausgewertet* |
| GLBPS = zahl | |
| LCLPS = zahl | |
| DFPPROC = *STD / *ADV[ANCED] | *wird nicht mehr ausgewertet* |
| GLBDFPN = zahl | *wird nicht mehr ausgewertet* |
| MAXDSBN = zahl | |

**LCLDFPS = zahl**
gibt die Größe der taskspezifischen Standard-ISAM-Pools in PAM-Seiten an.
Mögliche Werte: 96 ≤ zahl ≤ 8192
Standardwert: 512

**GLBPS = zahl**
legt die minimale Größe eines ISAM-Pools in einem Data-Space in PAM-Seiten fest. „zahl“ wird, wenn nötig, auf das nächste Vielfache von 512 gerundet und, wenn nötig, wieder auf den Maximalwert 32766 verringert.
Mögliche Werte: 512 ≤ zahl ≤ 32766
Standardwert: 512

**LCLPS = zahl**
legt den Standardwert für die Pool-Größe beim Anlegen von taskspezifischen ISAM-Pools per Kommando oder Makro in PAM-Seiten fest.
Mögliche Werte: 32 ≤ zahl ≤ 8192
Standardwert: 512

**MAXDSBN = zahl**
definiert die Anzahl der maximal für taskübergreifende ISAM-Pools bereitzustellenden Data-Spaces.
Mogliche Werte: 1 ≤ zahl ≤ 127
Standardwert: 2



Das Kommando SHOW-ISAM-CACHING informiert die Systembetreuung über die aktuell als ISAM-Caches verwendeten Data-Spaces, die zur Aufnahme von taskübergreifenden ISAM-Pools angelegt und verwaltet werden. Die Anzahl der verwendeten Data-Spaces kann für den laufenden Betrieb mit dem Kommando MODIFY-ISAM-CACHING geändert werden.

### Hinweis zur Einstellung von MAXDSBN und GLBPS

Mit den Voreinstellungen MAXDSBN=2 und GLBPS=512 können max. ca. 4000 gleichzeitig im Modus SHARUPD=YES geöffnete NK-ISAM-Dateien in separaten ISAM-Pools gepuffert werden. Die darüber hinaus im Modus SHARUPD=YES geöffneten NK-ISAM-Dateien werden in den bereits existierenden ISAM-Pools gepuffert.

Wenn sichergestellt werden soll, dass (fast) jede im Modus SHARUPD=YES geöffnete NK-ISAM-Datei separat gepuffert wird, muss zunächst ermittelt werden, wieviele NK-ISAM-Dateien maximal gleichzeitig mit SHARUPD=YES geöffnet werden. Wenn die Anzahl gleichzeitig geöffneter Dateien (=#FILES) den Grenzwert von 4000 überschreitet, muss die Anzahl der benötigten Data Spaces angepasst werden (mit Rundung auf das nächstgrößere Vielfache von 2 GB):

```
MAXDSBN = (#FILES x GLBPS) / 1.000.000
```

Anmerkung: Da die Poolgröße in Einheiten zu 2 KB angegeben wird, muss die Größe eines Data Space in 1.000.000 x 2 KB (= 2 GB) ausgedrückt werden.

Der Seitenwechselbereich ist ggf. entsprechend zu erweitern, wobei der Bedarf für Data Space für taskübergreifende ISAM-Pools mit `(MAXDSBN + 1) x 2 GB` anzusetzen ist.

### Ausschnitt aus der Parameterdatei

```
/BS2000 PARAMS
:
/BEGIN ISAM
LCLDFPS=100
GLBPS=512
LCLPS=512
MAXDSBN=4
/EOF
:
/END-PARAMS
```

## 3.9 Festlegung des TSN-Modus (JMS)

Mit der JMS-Anweisung in der Parameterdatei kann die Systembetreuung festlegen, ob im folgenden Systemlauf ausschließlich numerische Zeichen oder auch alphanumerische Zeichen für die Stellen 2 bis 4 einer Benutzer-TSN vom System vergeben werden.
Soll der Modus in zwei aufeinander folgenden Systemläufen von alphanumerisch nach numerisch gewechselt werden, kann die Systemeinleitung nur mit einem „Kaltstart“ durchgeführt werden. Der Operator erhält in diesem Fall einen entsprechenden Hinweis auf der Bedienstation.

Schlüsselwort für die Festlegung des TSN-Modus ist **JMS**.
Die Maximalzahl der dafür zugelassenen Parametersätze beträgt 16.

### Format des Parametersatzes zur Festlegung des TSN-Modus

| TSN-MODE = A[LPHANUMERIC] / N[UMERIC] |
|---|

**TSN-MODE = ALPHANUMERIC**
Es wird vereinbart, dass das System für die zweite bis vierte Stelle einer TSN auch alphanumerische Zeichen verwenden darf.
Wird in der Parameterdatei kein oder ein fehlerhafter Wert für die TSN-Vergabe gesetzt, dann wird dieser Modus für den Systemlauf wirksam.

**TSN-MODE = NUMERIC**
Es wird vereinbart, dass das System nur numerische Zeichen für die TSN einer Benutzertask vergeben darf. Folgt dieser Modus einem Systemlauf, in dem alphanumerische Zeichen vergeben wurden, kann die Systemeinleitung nur mit „Kaltstart“ durchgeführt werden.

### Ausschnitt aus der Parameterdatei

```
/BS2000 PARAMS
:
/BEGIN JMS
TSN-MODE=A ——————————————————————————————————————————————————— (1)
/EOF
:
/END-PARAMS
```

(1) Für den folgenden Systemlauf darf das System für die TSN einer Benutzer-Task für die Stellen 2 bis 4 auch alphanumerische Zeichen verwenden.

# 3.10 Speicherverwaltung (MEMORY)

Die Konfiguration der von der Systemfunktion Memory Management verwalteten Betriebsmittel Virtuelle Adressräume, Hauptspeicher und Erweiterungsspeicher kann bei der Systemeinleitung über entsprechende Einträge in der Parameterdatei definiert werden.



Die Parameter ES-UNIT, MEM-RECONF, HSDABSIZE, PAGING-GSSIZE werden nicht mehr ausgewertet. Ihre Angabe wird ignoriert.

Der Parameter SYSSIZE ist nicht mehr erforderlich. Siehe hierzu Hinweis auf Seite 103.

Schlüsselwort für den BEGIN-Satz des Parameterservice ist **MEMORY**.
Die Maximalzahl der dafür zugelassenen Parametersätze beträgt 16.

### Format des Parametersatzes zur Speicherverwaltung

| Parameter | |
|---|---|
| SHRSIZE n | |
| ASAMCTRL n | |
| ALAMCTRL n | |
| MEM-TEST n | *wird an SQ-Servern ignoriert* |
| SHXSIZE n | *wird an S-Servern ignoriert* |
| BIG-PAGE-QUOTA n | *wird an S- und SQ100-Servern ignoriert* |
| BIG-PAGE-SHRSIZE n | *wird an S- und SQ100-Servern ignoriert* |

**SHRSIZE n**

Größe des für Klasse-4-Speicher unterhalb der 16-MB-Grenze reservierten virtuellen Adressbereiches. Dieser Adressbereich wird für Shared-Code, DSSM-Subsysteme und TU-Logicals benötigt.

**n** ist die Größe des reservierten Bereichs in MB.
Die mit SHRSIZE reservierten Bereiche dürfen 8 MB nicht überschreiten.
Mögliche Werte: $1 \leq n \leq 8$
Standardwert: n = 2

**ASAMCTRL n**

Dieser Parameter dient zur Einstellung der Test and Trace Facility (TTF), die in ASAM eingebaut ist. Er legt die Größe des Klasse-3-Arbeitsspeichers, der für die Trace-Ausgabe zugewiesen wird, in 4KB-Blöcken fest. Die TTF dient nur zur Fehlerdiagnose und sollte nur im Bedarfsfall eingeschaltet werden.

**n** ist die Größe des TTF-Bereichs in 4KB-Blöcken.
Mögliche Werte: $0 \leq n \leq 15$
Standardwert: 0

**ALAMCTRL n**

Dieser Parameter dient zur Einstellung der Test and Trace Facility (TTF), die in ALAM eingebaut ist. Er legt die Größe des Klasse-3-Arbeitsspeichers, der für die Trace-Ausgabe zugewiesen wird, in 4KB-Blöcken fest. Die TTF dient nur zur Fehlerdiagnose und sollte nur im Bedarfsfall eingeschaltet werden.

**n** ist die Größe des TTF-Bereichs in 4KB-Blöcken.
Mögliche Werte: $0 \leq n \leq 15$
Standardwert: 0

**MEM-TEST n**

Dieser Parameter gibt an, ob bzw. wann ein Hauptspeichertest durchgeführt werden soll.

**n** bezeichnet die Option für den Hauptspeichertest. Mögliche Werte für n:

0 Beim Systemstart und während des Systemlaufs erfolgt kein Speichertest.
1 Der Speichertest soll beim Systemstart durchgeführt werden. Bei großen Arbeitsspeichern kann dies den Systemstart verzögern.
2 Vor jeder Neuvergabe einer Hauptspeicherseite an eine virtuelle Seite soll die Hauptspeicherseite getestet werden. Diese Option kann sich negativ auf die Systemperformance auswirken.

Standardwert: n = 0

Der Parameter wird an SQ-Servern ignoriert.

**SHXSIZE n**
Größe des Klasse-4-Speichers, der im Bereich oberhalb 16 MB und unterhalb 2GB angelegt wird. Dieser Speicherbereich wird von BLS und DSSM zum Laden von Shared Programmen und nichtprivilegierten Subsystemen, die im X86-64-Code vorliegen, genutzt. Der Bereich ist in allen Adressräumen eingebettet und kann im nichtprivilegierten Modus gelesen, aber nicht beschrieben werden.

**n** ist die Größe des Bereichs in MB und muss ein Vielfaches von 16 sein.
Mögliche Werte für n: 16, 32, 48, ..., 128
Standardwert: n = 64

Werden andere Werte eingegeben, wird die Systemeinleitung abgebrochen.

Der Parameter wird an S-Servern ignoriert.

**BIG-PAGE-QUOTA n**
Mit diesem Parameter wird die Zielgröße des Anteils am Hauptspeicher, der für „Big Pages“ reserviert sein soll, festgelegt. Eine Big Page hat eine Größe von 4 MB und wird für CISCFW-Kompilate genutzt.

**n** ist die Größe des für Big Pages vorgesehenen Hauptspeicher in %.
Mögliche Werte: $1 \leq n \leq 99$
Standardwert: n = 40

Der Parameter wird an S- und SQ100-Servern ignoriert.



Es ist nicht immer möglich, den angegebenen Prozentsatz in Big Pages anzulegen. Dies hängt von verschiedenen Randbedingungen ab (siehe Abschnitt „Big Pages für CISCFW-Kompilate (SQ-Server)“ auf Seite 131).

**BIG-PAGE-SHRSIZE n**
Mit diesem Parameter wird die Größe der „Shared Big Pages“ festgelegt, die im Shared-Memory (Klasse 3) angelegt und für CISCFW-Kompilate von Shared-Programmen genutzt werden.

**n** ist die Größe der Shared Big Pages in MB und muss ein Vielfaches von 16 sein.
Mögliche Werte: $0 \leq n$
Standardwert: n = 64

Der Parameter wird an S- und SQ100-Servern ignoriert.

*Hinweise zum Parameter SYSSIZE (Größe des Systemadressraumes auf S-Servern)*

Der Parameter ist seit BS2000/OSD V5.0 nicht mehr erforderlich, da der für den Systemadressraum zur Verfügung stehende Bereich sich aus der Größe des Benutzeradressraums und der sich daraus ableitenden Größe des Gesamtadressraums errechnet.

Ist trotzdem ein Wert für SYSSIZE im MEMORY-Parametersatz angegeben, so wird er auf Konsistenz mit dem errechneten Wert für SYSSIZE überprüft.

Mögliche Werte: 64, 80, 96, ..., 512 MB (Vielfaches von 16).
Wurden andere Werte eingegeben, wird die Systemeinleitung abgebrochen.

- Ist ein Wert für SYSSIZE angegeben und kleiner als der errechnete Wert, wird der angegebene Wert ignoriert und mit dem errechneten Wert gearbeitet.
- Ist ein Wert für SYSSIZE angegeben und der sich ergebende Wert für den Gesamtadressraum ist größer als 2 GB, wird die Systemeinleitung mit einer entsprechenden Meldung abgebrochen.
- Ist ein Wert für SYSSIZE angegeben und hat eine Verdopplung des Gesamtadressraums zur Folge, wird die Systemeinleitung mit dem erhöhten Wert für den Gesamtadressraum fortgesetzt. Es wird eine entsprechende Meldung ausgegeben, die auf die Inkonsistenz zwischen Generierung und Parameterdatei hinweist.

## 3.11 Konfiguration und Meldungsunterdrückung an Bedienstationen (OPR)

Die Funktion OPR-Tabellen-Parametrisierung gibt der Systembetreuung die Möglichkeit, die Bedienstationskonfiguration, die durch die Tabellen des Bediensystems logisch repräsentiert wird, erst während der Systemeinleitung zu ändern.

Die Änderungen betreffen folgende Punkte:

- Aufnahme von Operatorkommandos in die Operatorkommando-Tabelle
- Festlegung der Bedienstationskonfiguration (bei SKP-Systemen)
- Änderung der Bedienstationseigenschaften
- Änderung der Berechtigungsschlüssel für Operatorkommandos
- Zuordnung von Berechtigungsschlüsseln zu Bedienstationen
- Zuordnung von Filterstufen zu Bedienstationen (gilt nicht für berechtigte Anwendungen und im Fall von NBCONOPI=Y nur bis „System Ready“ und nach Shutdown)
- Unterdrückung bestimmter Meldungen an Bedienstationen (gilt nicht für berechtigte Anwendungen und im Fall von NBCONOPI=Y nur bis „System Ready“ und nach Shutdown)

Zum Zeitpunkt der Eingabe eines Operatorkommandos muss dieses, um ausgeführt werden zu können, dem Subsystem OPR bekannt sein. Zu diesem Zweck wird eine Kommandotabelle im Speicher aufgebaut, in der für jedes Kommando ein Eintrag enthalten ist. Dieser Eintrag besteht aus:

- Name des Kommandos
- Aliasnamen (Abkürzungen) des Kommandos
- Notwendige Berechtigung zur Kommandoausführung
- Identifikation des Kommando-Servers
- Kennwort-Indikator

Die Systembetreuung hat durch die dynamische Anpassung bzw. Erweiterung im Parameterservice (Anweisungen ADD-CMD-ENTRY, SET-CMD-CODE) die Möglichkeit, neue Kommandos in die Operatorkommando-Tabelle einzubringen bzw. Berechtigungsschlüssel für bestehende Kommandos zu ändern.

Semantische Fehler innerhalb eines Parametersatzes, die von der Fehlerroutine des Startup nicht erkannt und im Fehlerdialog behoben werden könnten, werden an der Bedienstation gemeldet.

Schlüsselwort für die Änderung oben genannter Punkte in der Parameterdatei ist **OPR**. Die Maximalzahl der dafür zugelassenen Parametersätze beträgt 256, dabei dürfen die einzelnen Parametersätze jeweils in einer maximalen Anzahl vertreten sein (siehe folgende Tabelle).

| Anweisung | Bedeutung | max. Anzahl |
|---|---|---|
| ADD-CMD-ENTRY | Operatorkommando in die Kommando-Tabelle aufnehmen | 100 |
| DEFINE-CONSOLE | Bedienstationen definieren oder ändern, Ersatz-Bedienstationen zuweisen | 24 |
| SET-CMD-CODE | Berechtigungsschlüssel eines Operatorkommandos ändern | 100 |
| SET-CODE | Berechtigungsschlüssel vergeben | 40 |
| SET-FILTER | Filterstufen festlegen | 40 |
| SET-MSG-SUPPRESSION | Ausgabe von Meldungen unterdrücken | 50 |

Tabelle 5: Übersicht über die Anweisungen zum Parametersatz OPR

Ist das Operator-Logon eingeschaltet (NBCONOPI=Y), wird die Bedienung an physikalischen Konsolen und an $CONSOLE-Anwendungen mit dynamischen Berechtigungsnamen in einem (im Sinne von Datensicherheit und Datenschutz) sicheren Modus durchgeführt.

Bei NBCONOPI=Y ist, ebenso wie bei NBCONOPI=N, in den Anweisungen SET-FILTER und SET-MSG-SUPPRESSION die Zuweisung an physikalische Konsolen durch die Angabe CONSOLE=*IPL möglich. Zuweisungen über mnemotechnische Gerätenamen – auch über die MN der IPL-Konsole – werden dagegen ignoriert.

Wurden Angaben in den Anweisungen SET-CODE, SET-FILTER und SET-MSG-SUPPRESSION ignoriert, erscheint folgende Meldung:

```
NBR1202 '(&00)' PARAM SETS FOR CONSOLES ONLY ALLOWED FOR *IPL
```

Der nachfolgende Ausschnitt aus der Parameterdatei zeigt die für die Änderung der Bedienstationskonfiguration relevanten Anweisungen bei NBCONOPI=N :

### Ausschnitt aus der Parameterdatei

```
/BS2000 PARAMS
:
/BEGIN OPR
DEFINE-CONSOLE CONSOLE=C1,REPLACEMENT=C2 ——————————————————————————————   (1)
SET-CODE CODE=A,CONSOLE=(C3,C4) ———————————————————————————————————————   (2)
SET-FILTER 3,CODE=*ALL,CONSOLE=*IPL ———————————————————————————————————   (3)
SET-FILTER *ALL,CODE=*ALL,CONSOLE=C5 ——————————————————————————————————   (4)
SET-MSG-SUPPRESSION (MSG0001,MSG0002),*ALL ————————————————————————————   (5)
/EOF
:
/END-PARAMS
```

(1) Der Bedienstation C1 wird die Ersatz-Bedienstation C2 zugewiesen.
War C1 schon durch die Hardware-Generierung vergeben, werden die Eigenschaften von C1 damit geändert.
Wurde C1 noch nicht in den Konsoltabellen geführt, wird ein SKP-Platzhalter-Eintrag für C1 mit den angegebenen Eigenschaften belegt.
Der REPLACEMENT-Operand ist bei NBCONOPI=N von Bedeutung.
Bei eingeschaltetem Operator-Logon wird er nicht ausgewertet.

(2) Den Bedienstationen C3 und C4 wird der Berechtigungsschlüssel A zugewiesen.

(3) Für alle der IPL-Bedienstation zugewiesenen Berechtigungsschlüssel wird die Filterstufe 3 gesetzt. Dies bewirkt, dass alle über Berechtigungsschlüssel verteilten Meldungen mit einem Meldungsgewicht 40-59 an der IPL-Bedienstation nicht ausgegeben werden.

(4) Für alle der Bedienstation C5 zugewiesenen Berechtigungsschlüssel werden die Filterstufen 1, 2, 3, 4 und 5 gesetzt. Dies bewirkt, dass alle über Berechtigungsschlüssel verteilten Meldungen unabhängig vom Meldungsgewicht (Fragen ausgenommen) an der Bedienstation C5 nicht ausgegeben werden. Diese Einstellung entspricht derjenigen, die mit dem Kommando ASR NOINF an der Bedienstation C5 vorgenommen werden könnte.
Die explizite Angabe eines mnemotechnischen Gerätenamens ist nur bei NBCONOPI=N möglich.

(5) Die Meldungen mit den Meldungsnummern `MSG0001` und `MSG0002` werden an allen in der Konsoltabelle geführten Bedienstationen unterdrückt.
Der Meldungsunterdrückungsmechanismus gilt bei eingeschaltetem Operator-Logon nur bis „System Ready“.

### 3.11.1 Anweisung ADD-CMD-ENTRY

**Format der Anweisung zum Aufnehmen von Operator-Kommandos in die Kommando-Tabelle**

| Anweisung | Operanden | Bedeutung |
|---|---|---|
| ADD-CMD-ENTRY | AUTHORIZATION-CODE = bs | Berechtigungsschloss für das Kommando |
| | ,CMD-NAME = kmd | Langform des Kommandonamens |
| | ,SAME-NAME = *NONE / kmd1..3 | Vergabe von Aliasnamen |
| | ,APPLICATION-NAME = name | Vereinbarung der zuständigen $CONSOLE-Anwendung |
| | ,PASSWORD-POSSIBLE = NO / YES | Vereinbarung der Kennwortprotokollierung |



Die Funktionalität von ADD-CMD-ENTRY kann von berechtigten Benutzerprogrammen auch mit dem Kommando CONNECT-CMD-SERVER abgedeckt werden.

**AUTHORIZATION-CODE = bs**
Berechtigungsschloss des Kommandos (ein Zeichen aus der Menge A-Z, 0-9, *, #, @, $). Mit diesem Berechtigungsschloss wird festgelegt, über welchen passenden Schlüssel (in Form des gleichlautenden Zeichens, auch Routing-Code genannt) ein Kommandogeber verfügen muss, um das Kommando nutzen zu können. Für Sonderfunktionen der Berechtigungsschlüssel @, $ und * siehe Tabelle 49 auf Seite 642.

**CMD-NAME = kmd**
Kommandoname (Original- bzw. Langform), für den ein Eintrag in der Operatorkommando-Tabelle erzeugt werden soll. Der Name kann bis zu 30 Zeichen lang sein und muss den Konventionen für Kommandonamen entsprechen
(in SDF-Syntax: <structured-name 1..30>).

**SAME-NAME =**
Legt fest, ob und welche Aliasnamen (d.h. zulässige Abkürzungen) für das Kommando definiert werden sollen.

**SAME-NAME = *NONE**
Voreinstellung: Es sollen keine Aliasnamen vereinbart werden.

**SAME-NAME = kmd1..3**
Menge der Aliasnamen (max. 3), die mit dem Kommandonamen gleichzusetzen sind.

**APPLICATION-NAME = name**
Name der kommandoausführenden $CONSOLE-Anwendung. Sofern noch nicht geschehen, wird die logische Konsole mit diesem Berechtigungsnamen automatisch intern generiert. Der Anwendungsname muss aus vier alphanumerischen Zeichen aus der Menge A-Z, 0-9, @, $, # bestehen, wobei das erste Zeichen keine Ziffer und nicht „#“ sein darf. Das erste Zeichen sollte ferner nicht „@“ sein, da dieses Zeichen für dynamische Berechtigungsnamen verwendet wird.

**PASSWORD-POSSIBLE =**
Vereinbart, ob das zu definierende Kommando möglicherweise ein Kennwort beinhaltet (z.B. Angabe eines Dateikennwortes in einem Operanden).
Ist ein solches Kennwort Bestandteil der Kommando-Zeichenfolge, kann mittels dieses Operanden verhindert werden, dass es in der System-Loggingdatei CONSLOG protokolliert wird, falls ein entsprechender SDF-Eintrag existiert.

**PASSWORD-POSSIBLE = NO**
Voreinstellung: Die gesamte Kommando-Zeichenfolge enthält kein Kennwort.
Vor dem Eintrag in die CONSLOG-Datei soll die Zeichenfolge also nicht nach einem Kennwort durchsucht werden.

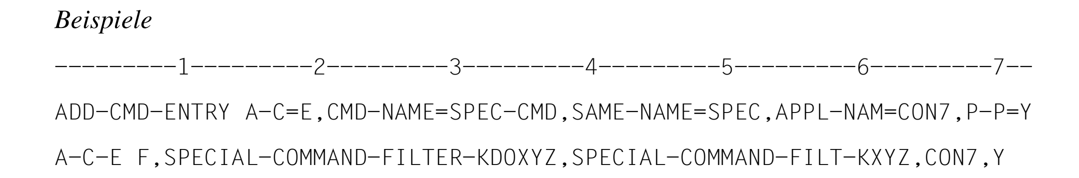
**PASSWORD-POSSIBLE = YES**
Die gesamte Kommando-Zeichenfolge kann Kennwörter enthalten. Um die Protokollierung dieses Kennwortes in der CONSLOG-Datei zu verhindern, wird die Systemtask CLOG zuständig für die CONSLOG-Datei) „befugt“, SDF aufzurufen und das Kennwort vor Aufnahme in die CONSLOG-Datei zu eliminieren.



In den ADD-CMD-ENTRY- und SET-CODE-Anweisungen sollten zusammen maximal 63 verschiedene logische Konsolen genannt werden. Bei Angabe von mehr als 63 logischen Konsolen werden nur die ersten 63 akzeptiert.

Die Länge einer Parameteranweisung ist generell auf 72 Zeichen beschränkt.
Es empfiehlt sich, wie folgende Beispiele zeigen, die Schlüsselwörter in einer Anweisung entweder stark abzukürzen oder ganz wegzulassen.
Der SAME-NAME muss dabei immer kürzer als der CMD-NAME sein.

*Beispiele*



```
---------1---------2---------3---------4---------5---------6---------7--

ADD-CMD-ENTRY A-C=E,CMD-NAME=SPEC-CMD,SAME-NAME=SPEC,APPL-NAM=CON7,P-P=Y

A-C-E F,SPECIAL-COMMAND-FILTER-KDOXYZ,SPECIAL-COMMAND-FILT-KXYZ,CON7,Y
```

## 3.11.2 Anweisung DEFINE-CONSOLE

**Format der Anweisung zur Definition oder Änderung von Bedienstationen und Zuweisung von Ersatz-Bedienstationen**

| Anweisung | Operanden | Bedeutung |
|---|---|---|
| DEFINE-CONSOLE | CONSOLE = name | Mnemotechnischer Gerätename der Bedienstation |
| | ,REPLACEMENT = STD /name | Mnemotechnischer Gerätename einer Ersatz-Bedienstation |
| | ,TELESERVICE = NO / YES | Teleservice-Eigenschaft |

**CONSOLE = name**
Mnemotechnischer Gerätename (2 Zeichen) der Bedienstation, für die die Zuweisungen gelten sollen. Der Name bezieht sich entweder auf eine bereits definierte Bedienstation oder auf eine mit diesem Namen zu definierende SKP-Bedienstation.
SKP-Bedienstationen können nur generiert werden, falls ein SKP generiert wurde; andernfalls erscheint die Meldung: `NBR0808 NUMBER OF CONSOLE MNEMONICS EXCEEDED`.

**REPLACEMENT =**
Mnemotechnischer Gerätename einer Bedienstation, die als Ersatz-Bedienstation verwendet werden soll. Die Angabe des Operanden wird bei NBCONOPI=Y ignoriert. Es wird folgende Meldung ausgegeben:
`NBR1203 OPERAND 'REPLACEMENT' IN 'DEFINE-CONSOLE' PARAMETER SET IGNORED`

**REPLACEMENT = STD**
Voreinstellung: Die Zuweisung der Ersatz-Bedienstation erfolgt durch die UCON-Task.

**REPLACEMENT = name**
Mnemotechnischer Gerätename einer durch die Hardware-Generierung oder durch einen Parametersatz definierten Bedienstation, die als Ersatz-Bedienstation verwendet werden soll. Gibt es die angegebene Bedienstation nicht, erfolgt die Zuweisung der Ersatz-Bedienstation durch die UCON-Task.

**TELESERVICE =**
Vereinbarung der Teleservice-Eigenschaft.

**TELESERVICE = NO**
Voreinstellung: Die Bedienstation soll keine Teleservice-Eigenschaft besitzen.

**TELESERVICE = YES**
Die Bedienstation soll Teleservice-Eigenschaft besitzen und kann nicht mehr Ersatzkonsole einer anderen Konsole und nicht mehr Hauptkonsole werden.

### 3.11.3 Anweisung SET-CMD-CODE

**Format der Anweisung zum Ändern des Berechtigungsschlosses eines Operatorkommandos**

| Anweisung | Operanden | Bedeutung |
|---|---|---|
| SET-CMD-CODE | AUTHORIZATION-CODE = bs | Berechtigungsschloss für das Kommando |
| | ,CMD-NAME = kmd | Langform des Kommandonamens |

**AUTHORIZATION-CODE = bs**
Neues Berechtigungsschloss des Kommandos (ein Zeichen aus der Menge A-Z, 0-9, *, #, @ oder $).
Mit diesem Berechtigungsschloss wird festgelegt, über welchen passenden Schlüssel (in Form des gleichlautenden Zeichens, auch Routing-Code genannt) ein Kommandogeber verfügen muss, um das Kommando nutzen zu können. Für Sonderfunktionen der Berechtigungsschlüssel @ und $ siehe Tabelle 49 auf Seite 642.

**CMD-NAME = kmd**
Kommandoname (Original- bzw. Langform), für den der Eintrag in der Operatorkommando-Tabelle geändert werden soll.
Der Name kann bis zu 30 Zeichen lang sein und muss den Konventionen für Kommandonamen entsprechen (in SDF-Syntax: <structured-name 1..30>).
Das zugeordnete Berechtigungsschloss (AUTHORIZATION-CODE) wirkt auch auf die dem Kommando zugeordneten Aliasnamen. Dies gilt auch für die mit der Anweisung ADD-CMD-ENTRY angemeldeten sog. Operator-Spezialkommandos.

### 3.11.4 Anweisung SET-CODE

**Format der Anweisung zur Vergabe von Berechtigungsschlüsseln**

| Anweisung | Operanden | Bedeutung |
|---|---|---|
| SET-CODE | CODE = bs / *ALL | (alle) Berechtigungsschlüssel |
| | ,CONSOLE = *IPL / name / (name,...) | Mnemotechnischer Gerätename der Bedienstation |

**CODE = bs**
Berechtigungsschlüssel (A..Z, 0..9, *,#,@,$), der der IPL-Bedienstation oder einer logischen Konsole mit generiertem Berechtigungsnamen zugeordnet wird.
Eine logische Konsole wird damit berechtigt, Operator-Kommandos einzugeben sowie Meldungen zu empfangen, die diesem Schlüssel zugeordnet sind.

**CODE = *ALL**
Den im CONSOLE-Operanden angegebenen Bedienstationen werden alle Berechtigungsschlüssel zugeordnet.

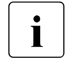
Bei NBCONOPI=Y:
Die mit der SET-CODE-Anweisung an logische Bedienstationen vergebenen Berechtigungsschlüssel berechtigen sowohl zum Meldungsempfang als auch zur Kommandoeingabe wie bei NBCONOPI=N.
Die der IPL-Konsole zugeordneten Berechtigungsschlüssel sind „implizite Routing-Codes“ (siehe Hinweise Seite 112). Ihre Wirkungsdauer kann mit dem Systemparameter NBIMPRCA festgelegt werden.

Bei NBCONOPI=N:
Die Bedienstationen werden damit berechtigt, alle Operator-Kommandos einzugeben und alle über einen Schlüssel verteilte Meldungen zu empfangen.

Die Schlüssel 0, *, $ und @ haben eine Sonderbedeutung (siehe auch Kapitel „Operatorfunktionen“ auf Seite 605).

**CONSOLE = (name,...,name)**
Mnemotechnischer Gerätename von logischen Bedienstationen (4 alphanumerische Zeichen) oder von physikalischen Bedienstationen (2 alphanumerische Zeichen), für die die Zuweisungen gelten sollen. Bei Angabe eines 4-stelligen Namens wird, sofern noch nicht geschehen, eine logische Konsole mit diesem Berechtigungsnamen automatisch intern generiert.
Das erste Zeichen des Berechtigungsnamens darf weder eine Ziffer noch das Zeichen „#“ sein, und es sollte auch nicht „@“ sein. Es empfiehlt sich jedoch, in den vier alphanumerischen Zeichen immer auch Sonderzeichen (#, @ oder $) anzugeben, um eine eindeutige Unterscheidung gegenüber Auftragsnummern (TSN) zu erreichen.

**CONSOLE = *IPL**
Die Zuweisung der Berechtigungsschlüssel soll für die Bedienstation gelten, von der das System geladen wird (=IPL-Konsole).



Folgendes gilt für NBCONOPI=Y:

– Berechtigungsschlüssel (Routing-Codes) werden in explizite und implizite Routing-Codes unterteilt.

– Explizite Routing-Codes werden mit dem Kommando REQUEST-OPERATOR-ROLE angefordert. Sie berechtigen zum Kommando- und Meldungsdialog.

– Implizite Routing-Codes werden vom Systembetreuer mit dem OPR-Parameterservice (Parametersatz SET-CODE) für die IPL-Konsole definiert. Sie können immer nur der IPL-Konsole oder der Hauptbedienstation zugeordnet werden und enthalten immer den Routing-Code *. Sie berechtigen nicht zum Kommandodialog.
  Implizite Routing-Codes berechtigen zum Meldungsdialog, also zum Empfang der über diese Routing-Code-Menge verteilten Systemmeldungen und zur Eingabe von Antworten auf Systemfragen.

– Sobald ein Routing-Code explizit einer Bedienstation zugeordnet wird (Kommando REQUEST-OPERATOR-ROLE), ist er nicht mehr in der Menge der impliziten Routing-Codes enthalten. Wird er wieder freigegeben (Kommando RELEASE-OPERATOR-ROLE), wird er wieder in die Menge der impliziten Routing-Codes aufgenommen.

Allgemein gilt:

– In den SET-CODE- und ADD-CMD-ENTRY-Anweisungen sollten zusammen maximal 63 verschiedene logische Konsolen genannt werden. Bei Angabe von mehr als 63 logischen Konsolen werden nur die ersten 63 akzeptiert.

### 3.11.5 Anweisung SET-FILTER

**Format der Anweisung zur Festlegung von Filterstufen**

| Anweisung | Operanden | Bedeutung |
|---|---|---|
| SET-FILTER | FILTER = *ALL / zahl / (zahl,...) | Filterstufe |
| | ,CODE = *ALL / bs / (bs,...) | Berechtigungsschlüssel |
| | ,CONSOLE = *IPL / name / (name,...) | Mnemotechnischer Name der Bedienstation |

Diese Anweisung wird bei NBCONOPI=Y nur für die IPL-Konsole zugelassen.
Die eingestellten Filterstufen wirken dann nur bis „System Ready“ und nach Shutdown.

**FILTER = *ALL**
Alle Filterstufen werden in den angegebenen Berechtigungsschlüsseln gesetzt. Damit werden die über die im CODE-Operanden angegebenen Berechtigungsschlüssel verteilten Meldungen unabhängig vom Meldungsgewicht an der IPL-Bedienstationen unterdrückt.

**FILTER = (zahl,...,zahl)**
Vereinbarung einer Filterstufe (Ziffer, Wertebereich 1...5), die zur Unterdrückung der über Berechtigungsschlüssel und Gewicht versandten Meldungen herangezogen wird.

**CODE = *ALL**
Der Filter soll für alle Berechtigungsschlüssel gesetzt werden.

**CODE = (bs,...,bs)**
Berechtigungsschlüssel (A..Z, 0..9, oder *,#,@,$), für die der Filter gesetzt wird.
Das Setzen von Filterstufen für den Berechtigungsschlüssel @ hat auf die Meldungsausgabe an Bedienstationen keinen Einfluss.

**CONSOLE = *IPL**
Die angegebenen Filter sollen für die Bedienstation gesetzt werden, von der das System geladen wird.
Bei NBCONOPI=Y wird dieser Wert aus Kompatibilitätsgründen nicht abgewiesen.

**CONSOLE = (name,,...name)**
Mnemotechnischer Gerätename (2 Zeichen) von Bedienstationen, an denen der Filter gesetzt werden soll.
Bei NBCONOPI=Y wird dieser Wert ignoriert.

## 3.11.6 Anweisung SET-MSG-SUPPRESSION

**Format des Parametersatzes zur Unterdrückung von Meldungsausgaben**

| Anweisung | Operanden | Bedeutung |
|---|---|---|
| SET-MSG-SUPPRESSION | MSG-ID = msg-id / (msg-id, ...) | Meldungsnummer |
| | ,CONSOLE = *ALL / *IPL / name / (name, ...) | Mnemotechnischer Gerätename der Bedienstation |
| | STARTUP-TYPE = ANY / DIALOG / NON-DIALOG | Abhängigkeit vom Startup-Typ |

Diese Anweisung wird bei NBCONOPI=Y nur für die IPL-Konsole zugelassen.
Die Meldungsunterdrückung wirkt dann nur bis „System Ready“.

**MSG-ID = (msg-id, ...)**
Angabe von 7-stelligen Meldungsnummern, deren Ausgabe auf der angegebenen Bedienstation unterdrückt werden soll.

**CONSOLE = *ALL**
Die vereinbarten Meldungen sollen an allen Bedienstationen unterdrückt werden.
Bei NBCONOPI=Y wird dieser Wert ignoriert.

**CONSOLE = *IPL**
Die vereinbarten Meldungen sollen an der Bedienstation unterdrückt werden, von der das System geladen wird.
Bei NBCONOPI=Y wird dieser Wert aus Kompatibilitätsgründen nicht abgewiesen.

**CONSOLE = name / (name, ...)**
Mnemotechnischer Gerätename (2 Zeichen) von Bedienstationen, an denen die Meldung unterdrückt werden soll.

**STARTUP-TYPE = ANY**
Die Meldungen sollen unabhängig vom Startup-Typ unterdrückt werden.

**STARTUP-TYPE = DIALOG**
Die Meldungen sollen nur bei DIALOG-Startup unterdrückt werden.

**STARTUP-TYPE = NON-DIALOG**
Die Meldungen sollen nur bei Nicht-DIALOG-Startup unterdrückt werden, also im FAST- oder AUTOMATIC-Startup.

## 3.12 Auswahl des Seitenwechselbereichs beim Startup (PAGING)

Die Systembetreuung kann über den Parameterservice die Datenträger bestimmen, die fürs Paging benutzt werden sollen. Dabei ist zu beachten, dass auf diesen Datenträgern Seitenwechseldateien (Paging-Dateien) mit dem Namen `SYS.PAGING.<vsn>` angelegt sind und dass im Startup nicht die max. mögliche Anzahl Paging-Dateien ausgenutzt werden kann (s.u.).

Der von Startup initialisierte Seitenwechselbereich muss eine Mindestgröße von 50 MB haben, sonst wird die Systemeinleitung mit der Meldung `NSI5225` abgebrochen.
Ist der Seitenwechselbereich größer als 50 MB, aber kleiner als 200 MB, wird als Warnung die Meldung `NSI5115` an der Konsole ausgegeben; die Systemeinleitung wird fortgesetzt.

Durch geeignete Auswahl der zum Seitenwechsel zu verwendenden Paging-Dateien kann die Seitenwechselaktivität bereits zum Startup-Zeitpunkt auf weniger belastete Datenträger gelenkt werden.
Es müssen nicht alle für Paging vorgesehene Datenträger zum Startup-Zeitpunkt verfügbar und verwendbar sein, da auch während des Systemlaufs zusätzliche Paging-Dateien in Betrieb genommen werden können.

Hat die Systembetreuung keine Parameter für die Auswahl von Paging-Dateien angegeben, dann werden automatisch alle auf dem Home-Pubset eingerichteten Paging-Dateien `SYS.PAGING.<vsn>`für den Seitenwechsel verwendet.

Mit dem Parameterservice können Datenträger mit Paging-Dateien innerhalb und außerhalb des Home-Pubsets angegeben werden.

*Einschränkungen*

– Zum Startup-Zeitpunkt muss mindestens eine Paging-Platte spezifiziert werden oder (ohne Parameterangaben) mindestens eine Paging-Datei auf dem Home-Pubset eingerichtet sein.

– Es dürfen maximal 128 Paging-Platten zum Startup-Zeitpunkt spezifiziert werden.

– Jede Paging-Datei `SYS.PAGING.<vsn>` kann aus mehreren Extents bestehen, die aber alle auf derselben Platte mit der Kennung vsn liegen müssen. Zusätzliche Extents verringern jedoch die Maximalzahl verwendbarer Paging-Dateien: Hat jede Datei z.B. 2 Extents, fasst die Startup-Tabelle nur noch 102 statt 128 Paging-Dateien; sind es 64 Dateien, dürfen diese max. 320 Extents haben.

– Die im Startup angegebenen Paging-Dateien dürfen sich auf max. 16 Pubsets verteilen.

– Kann keine Paging-Datei für eine spezifizierte Platte ermittelt werden, wird diese Platte ignoriert (Meldung `NSI5110`).

– Es können keine Privatplatten für Seitenwechsel verwendet werden.

– Paging-Pubsets müssen vollständig und exklusiv zur Verfügung stehen, dürfen also z.B. keine „shared Pubsets“ sein. Sie werden für den einzuleitenden Systemlauf vollständig und exklusiv von der Startup-Paging-Initialisierung belegt.
  Bei einem SM-Pubset wird nur der Volume-Set, der die Paging-Platten enthält, auf Fremdbelegungen überprüft.
– Die in der IPL-CONF abgespeicherte Gerätekonfiguration der Systemeinleitung wird bei neu angegebenen Paging-Platten automatisch um diese erweitert.

Schlüsselwort für den BEGIN-Satz des Parameterservice ist **PAGING**.

Die Maximalzahl dafür zugelassener Parametersätze beträgt 16. Überzählige Angaben werden ignoriert (Meldung `NSI0049`); die Systemeinleitung wird fortgesetzt. Wird während der Systemeinleitung in der PAGING-Anweisung ein Syntaxfehler diagnostiziert, wird sie – ohne Angabe eines Grundes – komplett abgewiesen.
An der Bedienstation kann die Anweisung entweder mit korrekten Werten wiederholt oder ignoriert werden.

### Format des Parametersatzes zur Auswahl des Seitenwechselbereichs (Paging-Area) zum Startup-Zeitpunkt

| Anweisung | Operand |
|---|---|
| PAGING | VOLUME = vsn / (vsn,...,vsn) |

**VOLUME =**
Angabe der Datenträger, die für den Seitenwechsel verwendet werden sollen. Die Datenträgerkennung VSN ist immer 6 Zeichen lang.

Folgende formale Regeln sind zu beachten:

– Mehrere identische Angaben werden nur einmal berücksichtigt.
– Es können max. 128 VSN-Angaben verarbeitet werden (16 Parametersätze mit je max. 8 VSNs). Überzählige Angaben werden ignoriert (Meldung `NSI0049`); die Systemeinleitung wird fortgesetzt.
– Werden alle angegebenen Paging-Anweisungen wegen Syntaxfehlern ignoriert, dann werden automatisch alle auf dem Home-Pubset eingerichteten Paging-Dateien `SYS.PAGING.<vsn>` für den Seitenwechsel verwendet.
– Für Seitenwechsel können in einem BS2000-Systemlauf grundsätzlich verschiedene Plattentypen kombiniert und gleichzeitig genutzt werden (im Hinblick auf Performance siehe auch „Performance Handbuch“ [39]).

**Ausschnitt aus der Parameterdatei**

```
/BS2000 PARAMS
:
/BEGIN PAGING
PAGING VOLUME=(PUBA01,PUBA02,PUBB01,PUBB02,PUBB03) ———————————————————— (1)
PAGING VOLUME=ABC.01 —————————————————————————————————————————————————— (2)
/EOF
:
/END-PARAMS
```

(1) Die Paging-Dateien (`SYS.PAGING.<vsn>`) auf den angegebenen Datenträgern der Pubsets mit den Katalogkennungen A und B werden Bestandteil des Seitenwechselbereichs (der Paging-Area) bei Startup.

(2) Auch die Paging-Datei auf dem angegebenen Datenträger des Pubsets mit der Katalogkennung ABC soll bei Startup zum Seitenwechselbereich zählen.

## 3.13 Snapshot-Initialisierung (SNAP)

SNAP sichert unter der Steuerung des eigenständigen, BS2000-unabhängigen SNAP-EXECs BS2000-spezifische Diagnoseinformationen (siehe die Beschreibung der SNAP-Funktion im „Diagnosehandbuch“ [14]). Dabei werden BS2000/OSD und dessen Anwendungen für max. 24 Sekunden (siehe Systemparameter „SNAPTIME“ auf Seite 758) angehalten.

Die Aktivierung der SNAP-Funktion erfolgt automatisch beim Startup von BS2000/OSD, wenn der Parameter SNAP-ACTIVE-SWITCH des SNAP-Parametersatzes nicht auf „OFF“ steht.

Schlüsselwort für den Snapshot-Parametersatz ist **SNAP**.
Es gibt für SNAP nur einen Parameter.

Wird ein ungültiger Wert im Parametersatz angegeben, wird der SNAP implizit ausgeschaltet (OFF) und eine Fehlermeldung beim Startup von BS2000/OSD ausgegeben.
Fehlt der Parameter SNAP-ACTIVE-SWITCH ganz, tritt die Voreinstellung (ON) in Kraft.

#### Format des Parametersatzes zur Snapshot-Initialisierung

| SNAP-ACTIVE-SWITCH = <u>ON</u> / ON␣ / OFF |
|---|

**SNAP-ACTIVE-SWITCH = <u>ON</u> / ON␣**
Voreinstellung: Die SNAP-Funktion soll für die laufende Session eingeschaltet werden.

Wenn die Systemdatei $TSOS.SNAPFILE nicht vorhanden ist, dann wird sie beim Startup automatisch (in Standardgröße) angelegt.

Wenn die SNAPFILE-Datei noch von einem SNAP-Dump der vorherigen Session beschrieben ist (Previous-SNAP), dann wird dieser SNAP-Dump in der neuen Session umkopiert, d.h. mit Indexstruktur versehen und in einer neuen Datei unter der Kennung SYSSNAP bereitgestellt.

**SNAP-ACTIVE-SWITCH = OFF**
SNAP wird bei der Systemeinleitung nicht initialisiert. Er steht für diese Session zunächst nicht zur Verfügung. SNAP-Aufrufe werden mit einem entsprechenden Returncode beendet. SNAP kann später dynamisch durch das Kommando ACTIVATE-SNAPSHOT aktiviert werden. Ein Previous-SNAP bleibt in der neuen Session unverändert erhalten.

Die Parameterwerte ON und OFF können auch von Hochkommas eingeschlossen werden ('ON' / 'ON␣' / 'OFF').

### Ausschnitt aus der Parameterdatei

```
/BS2000 PARAMS
:
/BEGIN SNAP
SNAP-ACTIVE-SWITCH=ON
/EOF
:
/END-PARAMS
```

## 3.14 Voreinstellung der Systemparameter (SYSOPT-CLASS2)

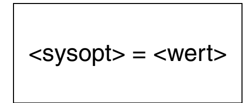
Es können nach dem Laden und der Rep-Verarbeitung des Klasse-2-Exec die Klasse-2-Systemparameter (kurz: Systemparameter) über den Startup-Parameterservice vorgegeben werden. Damit kann die Systembetreuung während der Systemeinleitung flexibel auf veränderte Vorgaben und Zielsetzungen reagieren.

Folgende Einschränkungen müssen berücksichtigt werden:

- Das Löschen oder Hinzufügen von Systemparametern ist nicht zulässig.
- Das Ändern bestimmter Attribute der Optionen (z.B. Länge der erlaubten Werte) ist nicht möglich.

Schlüsselwort für die Änderung von Systemparametern ist **SYSOPT-CLASS2**.
Die Maximalzahl der dafür zugelassenen Parametersätze beträgt 128.

#### Format des Parametersatzes zur Vorgabe von Systemparametern




**sysopt**
Bezeichnet den max. 8-stelligen Namen eines Systemparameters, dessen Wert verändert werden soll.
Eine Kurzbeschreibung aller Systemparameter mit Typ-Zugehörigkeit und Wertebereich ist im Anhang ab Seite 732 zu finden.

**wert**
Bezeichnet den Wert aus einem für den angegebenen Systemparameter gültigen Bereich. Wird einer Systemoption ein ungültiger Wert zugewiesen, dann wird die SYSOPT-Anweisung zurückgewiesen. Wird für eine Systemoption mehrfach ein gültiger Wert verwendet, dann wird nur der zuletzt angegebene gültige Wert berücksichtigt.
Gültig ist eine Anweisung dann, wenn der Typ und der zugewiesene Wert bzw. die Länge zur zugehörigen Systemoption passen.

Siehe auch den Abschnitt „Systemparameter“ auf Seite 732.

Folgende Aufstellung zeigt für jeden **Typ** die spezifischen Interpretationsregeln:

– Systemparameter vom Typ C (Zeichenketten)

  Der zugewiesene Wert wird als Zeichenkette interpretiert. Soll die Zeichenkette Leerzeichen enthalten, muss der Wert mit vorangestelltem C in Hochkommas eingeschlossen sein, kann dann jedoch selbst keine Hochkommas enthalten. Die Zeichenkette wird linksbündig eingetragen und ggf. mit Leerzeichen aufgefüllt.
  Ist die angegebene Zeichenkette länger als die maximale Länge des Systemparameters, wird die SYSOPT-Anweisung zurückgewiesen.

– Systemparameter vom Typ A (Arithmetische Werte)

  Der zugewiesene Wert wird als positiver dezimaler Wert interpretiert. Ist für den angegebenen Systemparameter ein Minimum und/oder Maximum definiert, wird der Wert daraufhin überprüft. Sonst wird als Minimum 0 angenommen und das Maximum aus der generierten Länge berechnet (siehe Längentabelle).

– Systemparameter vom Typ X (Hexadezimale Werte)

  Der zugewiesene Wert wird als Hexadezimalwert interpretiert, darf also nur aus Hexadezimalziffern (mit vorangestelltem X und in Hochkommas eingeschlossen) bestehen. Die Überprüfung auf Minimum und Maximum erfolgt wie bei Typ A aus der generierten Länge.

Für Typ A kann die generierte Länge 1, 2 oder 4 Byte betragen, für Typ X 1, 2, 4 oder 8 Byte. Ist kein Maximum definiert, werden folgende Maximalwerte akzepiert:

**Längentabelle**

| Typ | Länge =1 Byte | Länge = 2 Byte | Länge = 4 Byte | Länge = 8 Byte |
|---|---|---|---|---|
| A | 255 | 65535 | 2147483647 | - |
| X | FF | FFFF | FFFFFFFF | FFFFFFFFFFFFFFFF |

### Ausschnitt aus der Parameterdatei

```
/BS2000 PARAMS
:
/BEGIN SYSOPT-CLASS2
BMTNUM=32 ——————————————————————————————————————————————————— (1)
ENCRYPT=Y ——————————————————————————————————————————————————— (2)
EAMSPVS=X'01' ——————————————————————————————————————————————— (3)
TEMPFILE=C'#' ——————————————————————————————————————————————— (4)
/EOF
:
/END-PARAMS
```

(1) Die Anzahl der Ein-/Ausgabe-Puffer für die Katalogverwaltung wird auf 32 festgelegt.
Typ des Systemparameters: A.

(2) Es wird vereinbart, dass das System im folgenden Systemlauf mit verschlüsselten Kennwörtern arbeiten soll.
Typ des Systemparameters: C.

(3) Diese Anweisung steuert die Behandlung der SYSEAM-Datei auf einem Shared-Pubset.
Typ des Systemparameters: X.

(4) Als Sonderzeichen, das als Präfix bei temporären Dateien und Jobvariablen vor den Datei- bzw. JV-Namen gesetzt werden muss, wird für den nächsten Systemlauf „#“ vereinbart.
Typ des Systemparameters: C.

## 3.15 Änderung von IPL-Optionen (SYSOPT-IPL)

Mit dem Parametersatz SYSOPT-IPL kann die Systembetreuung das starre Prinzip „Flexibilität oder Komfort“ bei der Systemeinleitung, das in der Wahl des Modus (DIALOG oder FAST) entschieden wird, aufheben.

Durch den Modus DIALOG können Systembetreuer und Operator neue Software oder Korrekturen flexibel und im differenzierten Dialog mit dem Betriebssystem einsetzen. Allerdings verlängert sich dadurch die Ablaufzeit und durch die Beantwortung von Fragen muss auch eine erhöhte Gefahr von Fehleingaben in Kauf genommen werden.
Im komfortablen, dialogfreien Modus FAST hingegen kann wegen der Beschränkung auf Standarddateinamen nicht genügend flexibel auf veränderte Anforderungen (Einsatz neuer Software oder Korrekturen) reagiert werden.

Schlüsselwort für die Änderung von IPL-Optionen ist **SYSOPT-IPL**.
Die Maximalzahl der dafür zugelassenen Parametersätze beträgt 16.

Die Systembetreuung kann mit den Optionen einen Satz bestimmter Vorgaben für die Systemeinleitung auf die Verarbeitung der Parameterdatei auslagern und somit einen Mittelweg zwischen den Modi FAST und DIALOG einschlagen.
Die Anweisungen vom Typ SYSOPT-IPL werden unmittelbar vor dem Laden des BS2000-Klasse-1-Execs eingelesen und verarbeitet (zum genauen Ablauf der Systemeinleitung siehe Abschnitt „FAST-Startup“ auf Seite 44 und Abschnitt „DIALOG-Startup“ auf Seite 47).

### Format des Parametersatzes zur Änderung von IPL-Optionen

| **Format** | **Bedeutung** |
|---|---|
| BS2000 = datei | Dateiname für BS2000 |
| IOCF-CHECK = NO / zeit | Vorgaben zur Prüfung der IOCF-Identifikation |
| REPFILE1 = datei<br>:<br>REPFILE4 = datei | Dateiname für Klasse-1/2-Reps |
| LINKAGE-AUDIT-SCOPE =<br>NO / INTERRUPT-HANDLING / SYSTEMLEVEL | Vorgaben zur Aktivierung der Funktion LINKAGE-AUDIT |
| NEW-IPL-MODE = UNCHANGED / FAST | Angaben zur Änderung des Systemeinleitungsmodus |

**BS2000 = datei**
Mit dieser Anweisung wird der Dateiname für das zu ladende BS2000-Klasse-1-Exec und Klasse-2-Exec festgelegt. Ist diese Anweisung in der Parameterdatei belegt, dann wird diese Datei zum Laden der residenten (Klasse-1) und seitenwechselbaren (Klasse-2) Teile des Organisationsprogramms verwendet.

Kann die angegebene Datei nicht verarbeitet werden, so wird dies mit folgender Meldung protokolliert und ein ersatzweise zu verwendender Dateiname im Dialog erfragt:

```
NSI1192 INVALID FILENAME FOR BS2000 IN PARAMETER-SECTION SYSOPT-IPL
```

**IOCF-CHECK =**
Mit dieser Anweisung wird festgelegt, ob eine Überprüfung der Identifikation (= Zeitpunkt der Generierung) des aktuellen IOCF stattfinden soll oder nicht. Die Systembetreuung kann mit der Anweisung sicherstellen, dass das beim Starten des Servers (IMPL) vom Operator ausgewählte IOCF (I/O-Konfiguration für die Hardware) genau das zu einem bestimmten Zeitpunkt erstellte ist.

**IOCF-CHECK = NO**
Voreinstellung: Die Identifikation des aktuellen IOCF wird nicht geprüft.

**IOCF-CHECK = zeit**
Es soll geprüft werden, ob der im I/O-CONFIGURATION-REPORT des IOGEN protokollierte Generierungszeitpunkt des aktuellen IOCF mit der vorgegeben Zeitangabe übereinstimmt.
Der Wert <zeit> ist in der Form `yyyy-mm-dd hh:mm:ss` einzugeben. Wenn die geforderte Prüfung negativ ausfällt, wird dies mit folgender Meldung angezeigt:

```
NSI5206 IOCF-CHECK FAILED. GEN-TIME REQ.=yyyy-mm-dd hh:mm:ss
        IS=yyyy-mm-dd hh:mm:ss
```

**REPFILE[1..4] = datei**
Mit den Anweisungen REPFILE1, REPFILE2, REPFILE3 und REPFILE4 werden die Dateinamen der maximal vier möglichen Dateien für die zu verarbeitenden Reps (Objektcodekorrekturen) von BS2000/OSD festgelegt. Ist diese Anweisung in der Parameterdatei belegt, werden die angegebenen Dateien in allen Modi in der Reihenfolge REPFILE1 bis REPFILE4 für die Systemeinleitung zum Laden der Korrekturen verwendet.
Können die angegebene Datei oder die Dateien nicht verarbeitet werden, so wird dies mit folgender Meldung protokolliert und ein ersatzweise zu verwendender Dateiname im Dialog erfragt:

```
NSI1192 INVALID FILENAME FOR REPFILEn IN PARAMETER-SECTION SYSOPT-IPL
```

**LINKAGE-AUDIT-SCOPE =**
Mit dieser Anweisung kann gesteuert werden, ob die AUDIT-Funktionseinheit Linkage-AUDIT die Aufzeichnung von Sprungzieladressen nach Aufruf der Befehle BASR, BALR und BASSM sofort nach Laden des Klasse-1-Execs für einen zu bestimmenden Gültigkeitsbereich eingeschaltet werden soll.
Linkage-AUDIT kann auch zur Diagnoseunterstützung von Problemen genutzt werden, die in einer frühen Phase der Systemeinleitung auftreten können (prozessorlokaler Linkage-AUDIT).
Ist der Systemparameter AUDALLOW=NO eingestellt, werden die AUDIT-Funktionen bei Abarbeitung der Systemparameter für die Dauer des Systemlaufs ausgeschaltet.
Bei Bedarf kann die Linkage-AUDIT-Funktion nach „System Ready" auch mit dem Kommando STOP-LINKAGE-AUDIT wieder ausgeschaltet werden.

**LINKAGE-AUDIT-SCOPE = NO**
Voreinstellung: Die Protokollierung wird nicht eingeschaltet.

**LINKAGE-AUDIT-SCOPE = INTERRUPT-HANDLING**
Die Aufzeichnung der Sprungzieladressen wird nur für den SIH-Teil von BS2000/OSD (Funktionszustand SIH) eingeschaltet.

**LINKAGE-AUDIT-SCOPE = SYSTEM-LEVEL**
Die Aufzeichnung der Sprungzieladressen wird für den gesamten privilegiert ablaufenden Teil von BS2000/OSD (Funktionszustände TPR und SIH) eingeschaltet.

**NEW-IPL-MODE =**
Mit dieser Anweisung kann festgelegt werden, ob der vom Operator ausgewählte Modus der Systemeinleitung nach Einlesen der Parameterdatei beibehalten oder auf den Modus FAST geändert werden soll.

**NEW-IPL-MODE = UNCHANGED**
Voreinstellung: Der zu Beginn der Systemeinleitung eingestellte Modus bleibt unverändert.

**NEW-IPL-MODE = FAST**
Falls die Systemeinleitung im Modus DIALOG begonnen wurde, wird sie nach Einlesen der Parameterdatei im dialogfreien Modus FAST fortgesetzt.

### Ausschnitt aus der Parameterdatei

```
/BS2000 PARAMS
:
/BEGIN SYSOPT-IPL
BS2000=$TSOS.SYSPRG.BS2.180.TEST1 ———————————————————————— (1)
REPFILE1=$TSOS.SYSREP.BS2.180.REP1 ——————————————————————— (2)
REPFILE2=$TSOS.SYSREP.BS2.180.REP2
NEW-IPL-MODE=FAST ———————————————————————————————————————— (3)
/EOF
:
/END-PARAMS
```

(1) Der Dateiname für das während der Systemeinleitung zu ladende BS2000-Klasse-1-Exec und Klasse-2-Exec lautet $TSOS.SYSPRG.BS2.180.TEST1.

(2) Es wird vereinbart, dass das System für den folgenden Systemlauf die Objektcodekorrekturen aus den Dateien $TSOS.SYSREP.BS2.180.REP1 bzw. $TSOS.SYSREP.BS2.180.REP2 einlesen soll.

(3) Falls die Systemeinleitung im Modus DIALOG begonnen wurde, soll sie nach Einlesen der Parameterdatei im dialogfreien Modus FAST fortgesetzt werden.

# 4 Speicherverwaltung

Das Kapitel beginnt mit einer Beschreibung des virtuellen Adressraums und der virtuellen Speicherklassen. Anschließend wird die Verwaltung von Big Pages beschrieben. Danach werden die verschiedenen Speichermedien vorgestellt und Maßnahmen zur Verhinderung von Sättigungszuständen aufgezeigt.

Die systeminterne Speicherverwaltung von BS2000/OSD umfasst folgende Aufgaben:

- Verwaltung des virtuellen Adressraumes des Systems und der Benutzer
- Steuern von Anforderungen zur Reservierung, Belegung und Freigabe von Seiten im
  - virtuellen Adressraum (Virtual Address Space)
  - Hauptspeicher (Main Memory)
  - Globalspeicher (Global Storage)
  - Seitenwechselspeicher (Paging Area)

  sowie Aufbau und Aktualisierung von Tabellen zur Verwaltung dieser Speicher
- Steuerung des Seitenwechsels
- Zeitliche Überwachung der Belegung von Seiten im Hauptspeicher
- Erkennen von Sondersituationen und Einleiten entsprechender Maßnahmen zur
  - Sättigungsbehandlung
  - Fehlerbehandlung

Die systeminternen Mechanismen und Funktionen bieten der Systembetreuung von BS2000/OSD eine Reihe von Schnittstellen und Einflussmöglichkeiten, die im Folgenden erläutert werden.

## 4.1 Virtueller Adressraum und virtuelle Speicherklassen

Der virtuelle Adressraum für System und Benutzer, der durch Hauptspeicher und Seitenwechselspeicher realisiert ist, wird in folgende 6 Klassen unterteilt:

- Klasse 1: residente Systemmodule (z.B Unterbrechungsanalyse, Pagingroutine, Task-Management, physikalische Ein-/Ausgabe); residente Tabellen
- Klasse 2: seitenwechselbare Systemmodule (z.B Makro- und Kommandoverarbeitung)
- Klasse 3: residente Tabellen, die dynamisch angelegt und wieder abgebaut werden (z.B. TCBs, PCBs, Adressumsetztabellen)
- Klasse 4: seitenwechselbare Tabellen, die dynamisch angelegt und wieder abgebaut werden (z.B. Job-Management-Tabellen, DVS-Tabellen,Tabellen für Datenfernverarbeitung); nachladbare Systemmodule und sog. Shared Code, der über DSSM geladen und entladen werden kann
- Klasse 5: seitenwechselbare Tabellen, die nur von der jeweiligen Task benötigt werden (z.B TFT, E/A-Bereiche, Tabellen zur Verwaltung von Prozeduren, Kennwort-Tabelle)
- Klasse 6: Benutzerprogramm, Common-Memory-Pools

Die Klassen 1 bis 4 sind privilegierter Systemadressraum, der für den Benutzer nicht adressierbar ist. Eine Ausnahme ist der Shared-Code im Klasse-4-Speicher, der im Allgemeinen auch von Benutzern gelesen und ausgeführt werden kann.

Die Klasse 5 ist überwiegend privilegierter Benutzeradressraum, der als Kommunikationsbereich zwischen Benutzer- und Systemadressraum fungiert und mit Ausnahme der Ein-/Ausgabe-Bereiche für den Benutzer nicht adressierbar ist.

Die virtuelle Adressraumaufteilung kann bei der Systemeinleitung gestaltet und damit an die installationsspezifischen Bedürfnisse angepasst werden. Die dazu benötigten Daten für die Speicherverwaltung (Memory Management) werden über den Parameterservice eingelesen (siehe Kapitel „Parameterservice“ auf Seite 75).

| **Kommando** | **Bedeutung** |
|---|---|
| ADD-USER | Eintrag im Benutzerkatalog erstellen |
| MODIFY-USER-ATTRIBUTES | Festlegung der max. Größe des Klasse-6-Speichers einer Benutzerkennung (ADDRESS-SPACE-LIMIT) |
| SHOW-USER-ATTRIBUTES | Ausgabe der max. erlaubten Größe des Klasse-6-Speichers einer Benutzerkennung (Feld ADDRESS-SPACE-LIMIT) |
| SHOW-MEMORY-POOL-STATUS | Ausgabe der aktuell im System angelegten Memory Pools und der jeweils angeschlossenen Tasks |
| **Makro** | **Bedeutung** |
| ALESRV | Programm an einen Datenraum anschließen oder die Verbindung auflösen |
| ALINF | Informationen über die Zugriffslisten ausgeben, mit denen Datenräume und deren Verbindungen verwaltet werden |
| CSTAT | Status von Speicherseiten eines Programms ändern |
| CSTMP | Lese-/Schreibstatus auf einen Memory Pool ändern |
| DISMP | Teilnahme an einem Memory Pool beenden |
| DSPSRV | virtuellen Adressraum für Datenadressierung erzeugen, erweitern, verkleinern oder löschen, Informationen über einen Datenraum ausgeben oder diesen freigeben |
| ENAMP | Memory Pool einrichten oder die Teilnahme an einem bestehenden Memory Pool ermöglichen |
| MINF | Informationen über die Speicherbelegung und Größe des Klasse-6-Speichers oder eines Memory Pools ausgeben |
| RELM | zusammenhängenden Speicherbereich des aufrufenden Programms freigeben |
| RELMP | (zusammenhängenden) Speicherplatz eines Memory Pools freigeben |
| REQM | (zusätzlichen) Speicherplatz für das aufrufende Programm anfordern |
| REQMP | (zusammenhängenden) Speicherplatz für einen Memory Pool anfordern |

Tabelle 6: Schnittstellenübersicht zur Verwaltung des virtuellen Adressraums

## 4.2 Hauptspeicher

Der Hauptspeicher (Arbeitsspeicher) umfasst residente Seiten und Seiten, die in Working-Sets und im Free-Pool verwaltet werden.
Working-Sets bestehen aus virtuellen Seiten, die aktuell zugreifbar sind, aber prinzipiell auf den Seitenwechselspeicher ausgelagert werden können.
Zum Free-Pool gehören Seiten, deren Inhalt entweder gar nicht mehr benötigt wird oder (zumindest vorübergehend) nur im Seitenwechselspeicher aufbewahrt werden soll.
Bei Bedarf wird der Free-Pool aus den Working-Sets aufgefüllt.

Im BS2000-Native-Betrieb und für jedes VM2000-Gastsystem wird Hauptspeicher > 2 GB unterstützt.



Das (einzige) system-interne System Working Set-Verfahren (SYS-WS) zur Hauptspeicher-Verwaltung ist im „Performance-Handbuch“ [39] beschrieben.

| Kommando | Bedeutung |
|---|---|
| ADD-USER | Anzahl der residenten Hauptspeicherseiten für Benutzerprogramme einer Kennung festlegen (RESIDENT-PAGES) |
| MODIFY-MEMORY-PARAMETERS | Prozentsatz des Hauptspeichers festelegn, der für Big-Page-Speicher verwendet wird, sowie ob und wann Big-Page-Speicher abgebaut werden darf |
| MODIFY-SYSTEM-BIAS | Anzahl der residenten Hauptspeicherseiten für Benutzerprogramme festlegen |
| MODIFY-USER-ATTRIBUTES | Anzahl der residenten Hauptspeicherseiten für Benutzerprogramme einer Kennung ändern (RESIDENT-PAGES) |
| SHOW-MEMORY-CONFIGURATION | Aktuelle Konfiguration des Hauptpeichers inkl. Angaben zum Big-Page-Speicher anzeigen |
| SHOW-SYSTEM-STATUS | Anzahl der residenten Hauptspeicherseiten für Benutzerprogramme anzeigen (INFORMATION=SYSTEM-PARAM) |
| SHOW-USER-ATTRIBUTES | Anzahl der residenten Hauptspeicherseiten für Benutzerprogramme einer Kennung anzeigen (RESIDENT-PAGES) |

Tabelle 7: Kommandoübersicht zur Hauptspeicherverwaltung

### 4.2.1 Big Pages für CISCFW-Kompilate (SQ-Server)

In der Regel wird der Hauptspeicher in 4-K-Einheiten verwaltet. Diese Einheiten werden Frames genannt und den virtuellen Seiten zugeordnet.
Unter einer (realen) „**Big Page**“ versteht man eine Zusammenfassung von hintereinanderliegenden Frames in einer Größenordnung von mehreren MB, wobei der erste Frame auf das entsprechende MB-Vielfache ausgerichtet ist.

BS2000/OSD unterstützt Big Pages auf SQ-Servern (jedoch nicht SQ100) in einer Größe von 4 MB, das entspricht 1024 Frames. Sie werden für CISCFW-Kompilate genutzt, weil diese auf Big Pages schneller ablaufen.

Benutzte Big Pages können von der CISCFW auf Verlangen zurückgeben werden (z.B. bei der Reduktion des Hauptspeichers eines VM2000-Gastsystems).

Zur Festlegung der gewünschten Anzahl von Big Pages stehen Parameter in der Startup-Parameterdatei und das Kommando MODIFY-MEMORY-PARAMETERS zur Verfügung. Das Kommando SHOW-MEMORY-CONFIGURATION gibt neben Angaben über die Größe des Hauptpeichers auch Angaben zum Big-Page-Speicher aus (Beispiel siehe Seite 133).

#### Big Pages bei der Systemeinleitung

Die Startup-Parameterdatei kann auch Parameter zur Anlage von Big Pages enthalten.

*Einstellungen in der Parameterdatei*

Im Parametersatz MEMORY wird Folgendes festgelegt:

- mit dem Parameter BIG-PAGE-SHRSIZE die Größe (in MB) des Big-Page-Speichers, der im Shared-Memory (Klasse-3-Speicher) angelegt und für CISCFW-Kompilate von Shared-Programmen genutzt werden soll
- mit dem Parameter BIG-PAGE-QUOTA, welcher Anteil (in %) vom Hauptspeicher für Big Pages reserviert werden soll (gewünschte Zielgröße)

*Gründe für das Nichterreichen der Zielgröße*

Es ist nicht immer möglich, den angegebenen Prozentsatz in Big Pages anzulegen, denn je mehr logische Maschinen (CPUs) es im System gibt, umso kleiner wird der mögliche Anteil von Big Pages.

*Güte und Gültigkeit der beim Startup festgelegten Werte*

Wieviel realer Big-Page-Speicher (realer Hauptspeicher der Big Pages) im System angelegt werden soll, wird beim Systemstart im MEMORY-Parameterservice (Parameter BIG-PAGE-QUOTA) festgelegt. Der dort angegebene Prozentsatz ist allerdings nur eine Sollgröße. Die Istgröße fällt u.U. geringer aus, weil die Anlage des Big-Page-Speichers an

die maximal mögliche Anzahl der logischen Maschinen (CPUs) des Systems gekoppelt ist. Je geringer die Mindestgröße, je mehr CPUs, desto höher ist die Wahrscheinlichkeit, dass die Istgröße des Big-Page-Speichers die Sollgröße nicht erreicht.

Die eingestellten Werte für die Arbeit mit Big Pages sind für die aktuelle Session solange gültig, bis sie entweder explizit mit dem Kommando MODIFY-MEMORY-PARAMETERS oder implizit durch bestimmte Systemeigenschaften und -zustände verändert werden. Dazu gehören die Vergrößerung oder Reduzierung des Hauptspeichers oder eine drohende Hauptspeichersättigung.

#### Anpassung der Big Pages im laufenden Betrieb

Big Pages können gleichmäßig über den Hauptspeicher verteilt werden, da sie auch oberhalb des Hauptspeicherminimums liegen dürfen. Hieraus ergibt sich automatisch, dass die Größe des Big-Page-Speichers nach einer Hauptspeicher-Rekonfiguration weiter der BIG-PAGE-QUOTA entspricht (sofern nicht zuvor schon eine ungleichmäßige Verteilung der Big Pages entstanden war, z.B. durch explizite Veränderung der eingestellten Werte).

Sonst ist eine Veränderung der Ist-Größe des Big-Page-Speichers während des Systemlaufs nur beschränkt möglich.

Das Kommando MODIFY-MEMORY-PARAMETERS bietet zwei Möglichkeiten:

1. Mit dem Operanden BIG-PAGE-QUOTA kann der Prozentsatz des Hauptspeichers, der für Big-Page-Speicher verwendet werden soll, verändert werden.

   – Eine Erhöhung dieses Prozentsatzes führt nicht während der Kommandoausführung zu einer Erhöhung des Big-Page-Speichers, sondern erst bei Vergrößerung des Hauptspeichers (MEMORY-SIZE) durch Hinzunahme von Arbeitsspeicher bei dynamischer Rekonfiguration im Trägersystem.

   – Eine Verkleinerung dieses Prozentsatzes kann nur dann während der Kommandoausführung zu einer Verkleinerung des Big-Page-Speichers führen, wenn REDUCE-BIG-PAGES auf ON-QUOTA-EXCESS eingestellt ist und ungenutzter Big-Page-Speicher vorhanden ist. Ggf. wird auch die Rückgabe genutzter Big Pages von der CISCFW verlangt.

2. Mit dem Operanden REDUCE-BIG-PAGES kann eingestellt werden, ob und wann Big-Page-Speicher abgebaut werden darf:

   *NO Big-Page-Speicher darf nie abgebaut werden.

   *ON-CORE-SATURATION
   Abbau, wenn Hauptspeichersättigung droht, die durch die Zerschlagung einer Big Page in normalen Arbeitsspeicher aber vermieden werden kann. Ggf. wird auch die Rückgabe genutzter Big Pages von der CISCFW verlangt.
   Dieser Wert ist beim Systemstart voreingestellt.

*ON-QUOTA-EXCESS

Abbau während bzw. in Folge des aktuell aufgerufenen Kommandos MODIFY-MEMORY-PARAMETERS, wenn bzw. sobald die Istgröße des Big-Page-Speichers die aktuell eingestellte Sollgröße unterschreitet.

Diese Situation kann in folgenden Fällen auftreten:

– Sie kann beim aktuellen Kommandoaufruf bereits vorliegen.
– Sie kann bei weiterer Herabsetzung des Prozentsatzes durch ein nachfolgendes Kommando MODIFY-MEMORY-PARAMETERS eintreten.
– Sie kann bei späterer Speicherreduktion eintreten.

Voraussetzung für eine Verringerung der Istgröße ist außerdem, dass unbenutzte Big Pages vorhanden sind, die abgebaut werden können. Wenn diese Voraussetzung erst später durch Freigabe von Big Pages bei Programmbeendigung geschaffen wird, so verzögert sich der gewünschte Abbau von Big-Page-Speicher entsprechend.

Die Werte *ON-QUOTA-EXCESS und *ON-CORE-SATURATION können kombiniert werden.

### Ausgaben der Hauptspeicher- und Big-Page-Verwaltung

Die Größe des Speicherbereiches für die Big Pages wird über die Meldung `EMM2309` an der Konsole ausgegeben. Diese Meldung erscheint auch, wenn sich an dieser Größe im laufenden Betrieb etwas ändert, z.B.:

```
EMM2309 THE SIZE OF THE BIG PAGE MEMORY IS 168 MB.
```

Mit dem Kommando SHOW-MEMORY-CONFIGURATION kann jederzeit eine Statusabfrage zu Zuteilung und Auslastung des gesamten Hauptspeichers und der Big-Page-Bereiche erfolgen.

*Beispiel*

```
/SHOW-MEMORY-CONFIGURATION

 REAL MEMORY MANAGEMENT REPORT ----------------------------------------  (1)

   MEM-SIZE  MIN-MEM-SIZE  FREE-CORE-SIZE  FREE-MEM-SIZE  PAGE-MEM-SIZE  CSL
    1024 MB        256 MB        32.25 MB       32.25 MB      800.25 MB  0

 BIG PAGE MANAGEMENT REPORT -------------------------------------------  (2)

   QUOTA   PLANNED  #LM    ACTUAL      FREE     SHR   FREE-SHR       REDUCE
      40    400 MB    4    168 MB    104 MB   64 MB      16 MB   CORE/QUOTA
```

(1) Im Beispiel hat das System einen Hauptspeicher von MEM-SIZE = 1024 MB, wobei seine Mindestgröße MIN-MEM-SIZE = 256 MB beträgt.
(In einem VM2000-Gastsystem kann MIN-MEM-SIZE kleiner als MEM-SIZE sein. Im Native-Betrieb sind die Werte derzeit gleich.)

FREE-CORE-SIZE ist der Anteil am freien Hauptspeicher, der noch für residente Klasse-3-Speicher-Anforderungen des Systems zur Verfügung steht. Diese Größe ist auch maßgebend für den Hauptspeichersättigungsgrad.
FREE-MEM-SIZE ist der Anteil am freien Hauptspeicher, der für die restlichen residenten Speicheranforderungen wie residente Memory-Pools oder Data Spaces zur Verfügung steht. (Bei Hauptspeicher ≤ 2 GB sind FREE-CORE-SIZE und FREE-MEM-SIZE identisch.)

PAGE-MEM-SIZE stellt die Größe des Hauptspeichers dar, der für nicht-residente, seitenwechselbare Seiten zur Verfügung steht.
(In Systemen, bei denen MEM-SIZE und MIN-MEM-SIZE identisch sind, sind auch FREE-MEM-SIZE und PAGE-MEM-SIZE identisch.)

Die letzte Spalte CSL gibt den aktuellen Hauptspeichersättigungsgrad an, dabei sind die Werte 0 (keine Sättigung), 1, 2 oder 3 (höchste Alarmstufe) möglich (siehe auch Seite 184).

(2) QUOTA gibt den Prozentsatz des Hauptspeichers an, der als Big-Page-Speicher verwendet werden soll (und der mit dem Parameter BIG-PAGE-QUOTA in der Parameterdatei bzw. mit einem vorangegangenen Kommando MODIFY-MEMORY-PARAMETERS eingestellt wurde).

Aus der Hauptspeichergröße (MEM-SIZE) und dem Prozentsatz (QUOTA) berechnet sich die Sollgröße, die bei PLANNED angezeigt wird.

#LM ist die Anzahl der max. möglichen logischen Maschinen (CPUs) des Systems, also nicht die zum Zeitpunkt des Kommandoaufrufes angeschlossenen (attached) LMs. Neben der Mindestgröße des Arbeitsspeichers spielt die Anzahl der LMs eine wesentliche Rolle für die aktuelle Größe des Big-Page-Speichers: je höher die Anzahl der LMs, desto kleiner kann die Größe des Big-Page-Speichers ausfallen.

Die Spalte ACTUAL gibt die aktuelle Größe des Big-Page-Speichers aus, d.h. seine Istgröße.
Diese kann von der geplanten Sollgröße (PLANNED) abweichen, wenn z.B. Big-Page-Speicher wegen drohender Hauptspeichersättigung abgebaut wurde.

FREE gibt die Größe des freien (aktuell nicht benutzten) Big-Page-Speichers aus.

Die Größe des vorgesehenen shared Big-Page-Speichers für CISCFW-Kompilate im Klasse-3-Speicher gibt die Spalte SHR aus und FREE-SHR die Größe des freien shared Big-Page-Speichers.

Eine wichtige Angabe findet sich in der letzten Spalte REDUCE: Hier steht, wann Big-Page-Speicher abgebaut werden darf. Im Beispielfall bei drohender Hauptspeichersättigung (CORE) und falls sich durch Änderung der Hauptspeichergröße oder von BIG-PAGE-QUOTA eine neue Sollgröße des Big-Page-Speichers ergibt, die kleiner als die Istgröße ist (QUOTA).

### Allgemeine Hinweise

- Für die Hauptspeicherverwaltung gelten generell die folgenden Bedingungen:
  - MEM-SIZE ≥ MIN-MEM-SIZE > FREE-MEM-SIZE ≥ FREE-CORE-SIZE
  - MEM-SIZE > PAGE-MEM-SIZE ≥ FREE-MEM-SIZE ≥ FREE-CORE-SIZE
- CISCFW-Speicher oder DAB-Puffer werden in dem mit FREE-MEM-SIZE bezeichneten Speicher angelegt.Trotzdem kann eine Hauptspeichersättigung durch Abbau von DAB-Puffern oder Terminierung von Programmen beseitigt werden, wenn FREE-MEM-SIZE minus FREE-CORE-SIZE kleiner als die Größe der DAB-Puffer und/oder des CISCFW-Speichers ist.
  Ist dies nicht der Fall, sind Maßnahmen zur Reduktion des Klasse-3-Speichers bzw. zum Abbau von Adressräumen (Tasks, Data Spaces) erforderlich.

### Hinweise für VM2000

- Nur in VM2000-Gastsystemen gibt es ein sog. Hauptspeicherminimum.
  Dieses Minimum, das von der Gesamtspeichergröße nach unten abweicht, ermöglicht es, eine Speicherreduktion des Systems durchzuführen.
- In einem VM2000-Gastsystem kann man der gewünschten Sollgröße des Big-Page-Speichers näher kommen, wenn man den Gesamtspeicher des Gastsystems vergrößert.
- Ist für ein laufendes VM2000-Gastsystem eine Speicherreduktion vorgesehen, dann ist für diese VM eine MIN-MEM-SIZE abweichend von der MEMORY-SIZE festzulegen. Um eine solche Speicherreduktion zu ermöglichen, befriedigt die Speicherverwaltung von BS2000/OSD alle Anforderungen von residentem Speicher, die unterhalb dieser MIN-MEM-SIZE liegen. Eine ausreichende Dimensionierung dieses Parameters ist daher notwendig. Siehe dazu auch den Abschnitt „Performance-Aspekte bei VM2000-Betrieb (Hauptspeicher)“ im „Performance-Handbuch“ [39].

## 4.3 Globalspeicher (S-Server)

Der Globalspeicher (GS) ist ein Erweiterungsspeicher auf Halbleiterbasis, der durch eine optionale Batterie nichtflüchtig wird. Der GS ermöglicht durch seine Nichtflüchtigkeit einen ausfallsicheren, schnellen, synchronen Zugriff auf die Daten.
Gegenüber Platten-Ein-/Ausgaben erhöht sich die Zugriffsgeschwindigkeit erheblich.
Die Ausfallsicherheit wird dadurch gewährleistet, dass BS2000/OSD neben den Nutzdaten auch alle Verwaltungsdaten im Globalspeicher führt, so dass nach einem Systemabbruch und Restart die Zuordnung der Daten bekannt und diese somit weiterhin verfügbar sind.

Der Globalspeicher wird mit maximal zwei unabhängigen Hardware-Einheiten (**GS-Units** genannt) ausgeliefert, die von BS2000/OSD aus angesprochen werden können. Jede GS-Unit ist ein selbstständiges Gerät mit optionaler Batterie. Die maximale Größe des GS ist abhängig vom jeweiligen Server-Typ, an den der GS angeschlossen wird, sowie von der Kapazität der im GS verwendeten Speicherbausteine.

Eine räumlich getrennte Aufstellung der beiden GS-Units ist möglich. Die Verbindungen zu den Servern sind Lichtwellenleiter in den Längen 20 Meter und 70 Meter. Die räumlich getrennte Aufstellung ermöglicht so auch einen Schutz gegen (begrenzte) Katastrophen.

#### GS-Ausfallsicherheit

BS2000/OSD bietet Schnittstellen zur dynamischen Rekonfiguration der GS-Units. Es werden Befehle zum gleichzeitigen Schreiben auf beide GS-Units (das sog. Duplicate-Write) unterstützt. Diese Funktion bietet zusätzlich eine Fehlererkennung (equivalency damage), die neben dem Performance-Gewinn (nur einer statt zwei Schreibaufträgen) auch eine erhöhte Ausfallsicherheit gewährleistet.

#### GS-Complex und Parallel HIPLEX

Ein GS kann von 2-4 Servern gleichen Typs gemeinsam genutzt werden. Die Server können auch gemischt an den gleichen GS angeschlossen werden.

Die Möglichkeit, verschiedene Server-Typen an den gleichen GS anzuschließen, hängt vom Hardware-Typ des GS ab und muss im Einzelfall technisch geklärt werden.

Ein solcher Hardware-Verbund wird als **GS-Complex** bezeichnet. Die beteiligten Server heißen **GS-Server**.

Ein GS-Complex bietet auch eine Kommunikationsfunktion über den GS an (GSIGP, GS-Signalprocessing), die im **Parallel HIPLEX** von BS2000/OSD zum schnellen Austausch globaler Locks genutzt wird. Für die gemeinsame Nutzung des GS durch mehrere BS2000-Systeme müssen diese einen XCS-Verbund bilden; man bezeichnet diesen Hardware- und Software-Verbund als „Parallel HIPLEX“.
Ausführliche Informationen zu HIPLEX finden Sie im Handbuch „HIPLEX MSCF“ [33].

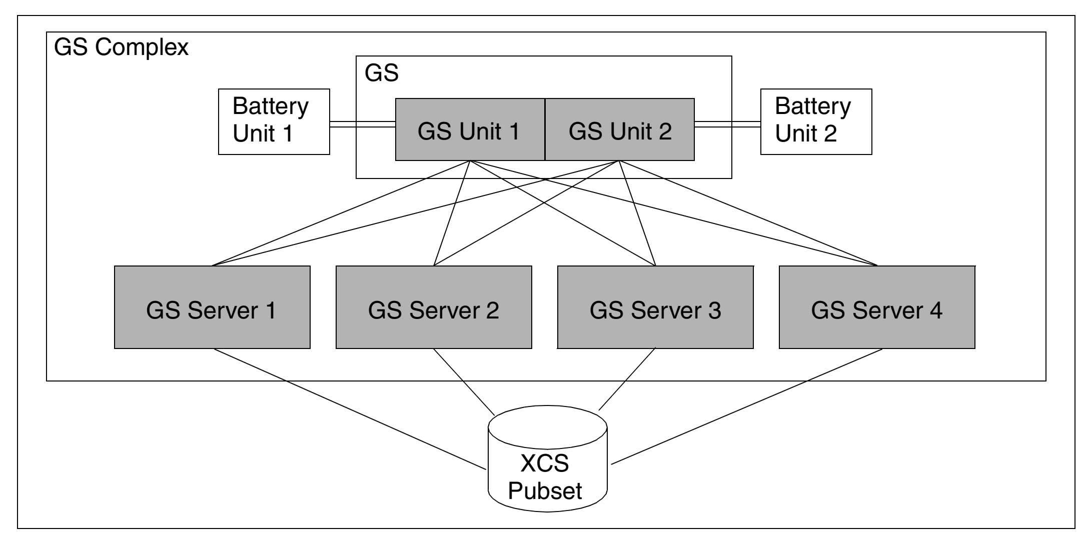



Bild 2: Parallel HIPLEX auf Basis eines GS-Complexes, bestehend aus 4 GS-Servern und 2 GS-Units

Unter VM2000 sind die Server eines Parallel HIPLEX die über realen oder virtuellen GS gekoppelten Gastsysteme. Mit virtuellem GS sind bis zu 15 Server in einem Parallel HIPLEX konfigurierbar. Bei der GS-Zuordnung sind zwei Szenarien zu unterscheiden:

- Neben Gastsystemen des eigenen Servers nehmen auch andere BS2000-Systeme auf anderen Servern am Parallel HIPLEX teil. In diesem Fall ist den Gastsystemen der reale GS zuzuordnen.
- Es nehmen nur Gastsysteme des VM2000-Systems am Parallel HIPLEX teil. In diesem Fall kann ein virtueller GS als Kopplungselement zugeordnet werden.

Die Verwaltung des GS in BS2000/OSD erfolgt durch das Subsystem GSMAN.

| **Kommando** | **Bedeutung** |
|---|---|
| *GS-Konfiguration* | |
| CREATE-GS-PARTITION | Erstellen einer Partition auf dem GS |
| COPY-GS-PARTITION | Kopieren der Daten einer Partition in eine zweite Partition |
| DELETE-GS-PARTITION | Löschen einer Partition auf dem GS |
| FORCE-DESTROY-GS-PARTITION | Löschen einer Partition auf dem GS erzwingen |
| SHOW-GS-STATUS | Anzeige des aktuellen Status der GS-Konfiguration |
| *GS-Unit-Rekonfiguration* | |
| ATTACH-GS-UNIT | Inbetriebnahme einer GS-Unit |
| DETACH-GS-UNIT | Wegschalten einer GS-Unit |
| MODIFY-GSMAN-PARAMETERS | Ändern des GSMAN-Parameters GS-USAGE |
| *GS-Complex-Rekonfiguration* | |
| CREATE-GS-COMPLEX | Erstellen eines neuen GS-Complexes bestehend aus dem eigenen GS-Server und der angegebenen GS-Unit |
| MODIFY-GS-COMPLEX | Ändern eines GS-Complexes durch Hinzufügen oder Wegnehmen von GS-Servern und GS-Units |
| SHOW-GS-COMPLEX-CONFIGURATION | Anzeige der aktuellen GS-Complex-Konfiguration |
| *GS-Server-Rekonfiguration* | |
| CONNECT-GS-SERVER | Konnektieren eines GS-Servers mit den GS-Units eines GS-Complexes |
| DISCONNECT-GS-SERVER | Diskonnektieren eines GS-Servers von den GS-Units eines GS-Complexes |
| *GS-Diagnose* | |
| START-GS-TRACE | Starten eines Diagnose-Traces auf dem GS |
| STOP-GS-TRACE | Beenden eines laufenden Diagnose-Traces |

Tabelle 8: Kommandoübersicht Globalspeicher

### 4.3.1 Das GS-Nutzungskonzept

Als Verwaltungs- und Nutzungseinheit wird der GS in Partitions eingeteilt.
In einem XCS-Verbund (Cross Coupled System) besteht die Möglichkeit, einen GS von mehreren Systemen aus gemeinsam zu nutzen (siehe dazu den Abschnitt „GS-Betrieb im XCS-Verbund“ auf Seite 149). Der GS ermöglicht dabei eine schnelle Kommunikation zwischen den beteiligten Systemen über die Funktion GSIGP.
Die gemeinsame Benutzung des GS durch mehrere BS2000-Systeme muss über die Subsystem-Parameter-Datei GS-USAGE=GLOBAL eingestellt werden; für die exklusive Nutzung durch BS2000/OSD ist hier der Wert GS-USAGE=LOCAL einzustellen.

Die Nutzung des Globalspeichers wird über verschiedene Ebenen realisiert:
**GS-Units - GS-Partition - Nutzungsbereiche** (z.B. Cache-Bereich).

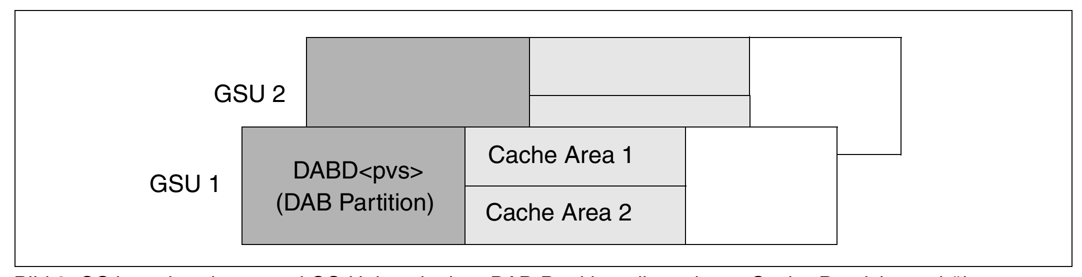



Bild 3: GS bestehend aus zwei GS-Units mit einer DAB-Partition, die mehrere Cache-Bereiche enthält

Neben DAB und VM2000 zählen auch GSVOL (Volume-Emulation, siehe Seite 171) und NSM (Node-Synchronisation-Manager) zu den Nutzern des GS.

Die Einteilung des GS in Partitions trifft die Systembetreuung mit Hilfe der GSMAN-Kommandos. Die Einteilung von Partitions in Nutzerbereiche erfolgt durch die Nutzer selbst und ist für GSMAN transparent. Die Namen der Partitions leiten sich aus den Namen der Subsysteme ab.

Es gibt zwei Modi für Partitions:

- Mono-Partitions
  bestehen aus einem zusammenhängenden Adressraum einer GS-Unit.

- Dual-Partitions
  bestehen aus je einem zusammenhängenden Adressraum auf beiden GS-Units, wobei deren Anfangsadresse um die HW-Konstante „Duplication-Boundary“ differieren.

Die Nutzer DAB, GSVOL und VM2000 können gleichzeitig mehrere Mono- und/oder Dual-Partitions nutzen.

Partitions können von einem Nutzer exklusiv oder – im XCS-Verbund – im Shared-Modus genutzt werden. GS-Sharing und die Kommunikation über den GS (mit GSIGP) ist mit max. 16 Partnern möglich (wird durch XCS festgelegt).

**Verfügbarkeitsaspekte bei der Nutzung des Globalspeichers**

Zwei GS-Units besitzen unabhängige Ausfallwahrscheinlichkeiten. Daher ist die Verfügbarkeit von Daten in Dual-Partitions erheblich höher als die von Daten in Mono-Partitions. Für ein hochverfügbares System ist deshalb der Einsatz von zwei GS-Units und die ausschließliche Nutzung von Dual-Partitions zu empfehlen.

Bei einem differenzierten Einsatz von Mono- und Dual-Partitions ist zu beachten, dass die Daten der Nutzer DAB (gilt nur für Schreib-Caches) und GSVOL gegen Hardware-Ausfälle zu schützen sind, denn es handelt sich dabei um Benutzerdaten und Pubset-Verwaltungsdaten.
Die Daten des Nutzers NSM sind (temporäre) Systemdateien; ihre Verfügbarkeit ist allerdings Voraussetzung für die Verfügbarkeit des laufenden Systems.

## 4.3.2 GS-Konfiguration

Die Verwaltungs- und Nutzungseinheit des GS ist die Partition, ein zusammenhängender Adressbereich in Vielfachen von 1MB auf einer GS-Unit (Mono-Partition) bzw. je ein zusammenhängender Adressbereich in Vielfachen von 1 MB auf beiden GS-Units gespiegelt (Dual-Partition). Nutzer von Partitions sind DAB, GSVOL, VM2000 und NSM.

- Das Software-Produkt DAB bietet eine deutlich beschleunigte Ein-/Ausgabe-Verarbeitung (unabhängig von den benutzten Platten-Subsystemen) durch Zwischenschaltung eines schnellen Speichermediums (Hauptspeicher und GS) als Cache für ausgewählte Ein-/Ausgaben. DAB erlaubt die Nutzung von GS-Partitions sowohl im exklusiven Modus wie im Shared-Modus.
- GSVOL bietet hochperformante Daten-Volumes (sog. emulierte GS-Volumes) an. Zusätzlich zu der exklusiven Nutzung von Partitions durch ein BS2000-System erlaubt GSVOL die gemeinsame Nutzung von GS-Volumes als Shared-GS-Volumes durch mehrere BS2000-Systeme in einem XCS-Verbund.
- NSM nutzt den GS zur performanten Kommunikation im XCS-Verbund.
- VM2000 nutzt Partitions als virtuellen GS für Gastsysteme.

Alle Nutzer verlangen die Beachtung bestimmter Namenskonventionen:

- DAB verlangt Partition-Namen in der Form „DABx<catid>“
  (mit x=D für Dual-Modus, x=1/x=2 für die GS-Unit 1 bzw. 2 im Mono-Modus)
- Partition-Namen von GSVOL müssen mit „GSV“ beginnen.
- NSM nutzt eine Partition mit dem Namen „NSM“.
- VM2000 nutzt die Partitions „VIRTGS01“ ... „VIRTGS16“.

Die max. Größen und Anzahl der Partitionen der einzelnen Nutzer sind unterschiedlich; die folgende Tabelle gibt einen Überblick. Zu beachten ist, dass in Summe max.1000 Partitionen angelegt werden können und dass eine Partition max. die Größe einer GS-Unit (weniger 2 MB) erreichen kann. Im Subsystem GSMAN ist eine maximale Größe einer Partition von 32 TB codiert; die Hardware unterstützt theoretisch 64 TB pro GS-Unit.

| Nutzer | max. Größe | Max. Anzahl |
|---|---|---|
| DAB | 32 TB | 1000 |
| GSVOL | 2 TB (Beschränkung durch DVS) | 256 |
| NSM | 4 MB | 1 |
| VM2000 | 32 TB | 16 |

### Anlegen einer Partition

Mit dem Kommando CREATE-GS-PARTITION erstellt die Systembetreuung eine Partition auf dem GS. Dieser wird dabei ein eindeutiger Name, eine Größe (in Vielfachen von 1 MB), ein Modus (Mono oder Dual) und im Mono-Modus die zugehörige GS-Unit zugeteilt. Optional kann der Beginn der Partition im Kommando festgelegt werden, ebenfalls in Vielfachen von 1 MB.

### Kopieren einer Partition

Mit dem Kommando COPY-GS-PARTITION kann eine VIRTGS-Partition in eine zweite VIRTGS-Partition kopiert werden.

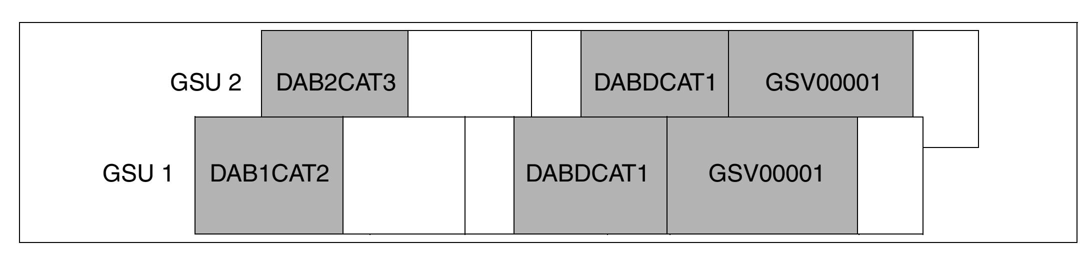
### Löschen einer Partition

Mit dem Kommando DELETE-GS-PARTITION kann die Systembetreuung eine Partition auf dem GS löschen. Es wird dabei vorausgesetzt, dass die Partition keine gültigen Daten enthält, andernfalls wird das Kommando abgewiesen.
Mit dem Kommado FORCE-DESTROY-GS-PARTITION kann in besonderen Situationen die Partition zwingend gelöscht werden.

Die Einteilung des GS in Partitions bleibt solange erhalten, bis Änderungen durch Kommandos vorgenommen werden oder der GS zurückgesetzt wird (durch RESET-SSU oder POWER OFF).

### Ablaufbeispiel einer GS-Konfiguration

Die Kommandos, die nach Laden des Subsystems GSMAN („GS-Manager“) durchgeführt werden müssen, sollten von der Systembetreuung in einer Enter-Datei hinterlegt werden. Zu beachten ist jedoch, dass eine erneute Konfiguration nur dann notwendig wird, wenn alle GS-Units gelöscht wurden (z.B. durch die SVP-Funktion RESET-SSU oder nach POWER-OFF). Im folgenden Beispiel wird angenommen, dass zwei GS-Units mit je 4 GB zur Verfügung stehen. Dabei ist CAT3 die Katalogkennung des Home-Pubsets.




```
/CREATE-GS-PARTITION PARTITION-ID=GSV00001,SIZE=1024,MODE=*DUAL   ———————  (1)
/CREATE-GS-PARTITION PARTITION-ID=DABDCAT1,SIZE=640,MODE=*DUAL  ————————  (2)
/CREATE-GS-PARTITION PARTITION-ID=DAB1CAT2,SIZE=640,MODE=*MONO(GS-UNIT=1) (3)
/CREATE-GS-PARTITION PARTITION-ID=DAB2CAT3,SIZE=640,MODE=*MONO(GS-UNIT=2) (4)
```

(1) Es wird eine Dual-Partition der Größe 1024 MB jeweils in der GS-Unit 1 und in der GS-Unit 2 für GSVOL angelegt. Diese GS-Partition ist in der Folge Behälter für ein GS-Volume mit doppelter Aufzeichnung.

(2) Es wird eine Dual-Partition der Größe 640 MB jeweils in der GS-Unit 1 und in der GS-Unit 2 für DAB angelegt. Diese GS-Partition ist in der Folge Behälter für einen DAB-Cache-Bereich mit doppelter Aufzeichnung.
In diesem DAB-Cache-Bereich können Daten des Pubsets mit der Katalogkennung CAT1 über PFA-Caching zwischengepuffert werden.

(3) Es wird eine Mono-Partition der Größe 640 MB auf der GS-Unit 1 für DAB angelegt. Sie ist in der Folge ein Behälter für einen DAB-Cache-Bereich mit einfacher Aufzeichnung.
In diesem DAB-Cache-Bereich können Daten des Pubsets mit der Katalogkennung CAT2 über PFA-Caching zwischengepuffert werden.

(4) Es wird eine Mono-Partition der Größe 640 MB auf der GS-Unit 2 für DAB angelegt. Sie ist in der Folge ein Behälter für einen DAB-Cache-Bereich mit einfacher Aufzeichnung.
In diesem DAB-Cache-Bereich können Daten beliebiger Platten über ADM-PFA-Caching zwischengepuffert werden.

Die Inbetriebnahme der Partitions als Nutzerbereiche geschieht über nutzerspezifische Kommandos.

### Ermitteln der aktuellen GS-Konfiguration

Die aktuelle GS-Konfiguration kann mit dem Kommando SHOW-GS-STATUS ermittelt werden. Die Ausgabe besteht je nach gewünschter Information aus bis zu vier Abschnitten:

1. Subsystem-Parameter GS-USAGE
2. GS-Unit-Daten
3. Partitions
4. freier Speicherplatz im GS

*Abschnitt 1: Subsystem-Parameter*

GS-USAGE=LOCAL/GLOBAL

*Abschnitt 2: GS-Unit-Daten*

GS-UNITS:

| GS-UNIT | BEGIN | END | SIZE | FREE | CONTIGUOUS | STATE |
|---|---|---|---|---|---|---|
| 1 | 0 | 4095 | 4096 | 1790 | 1278 | ATTACHED |
| 2 | 32768 | 36863 | 4096 | 1790 | 1278 | ATTACHED |

BEGIN und END geben den realen Adressraum der GS-Unit, SIZE zeigt die Größe der GS-Unit, FREE den aktuell freien Adressraum und CONTIGUOUS den größten zusammenhängenden freien Adressraum an. Alle Angaben sind in MB. Für die GSMAN-Verwaltungsdaten sind auf jeder der beiden GS-Units 2 MB fest reserviert.

*Abschnitt 3: Daten der Partitionen*

PARTITIONS:

| PART-ID | ACCESS | DATA | MODE | GSU | SIZE | BEGIN | END | ATT-DUAL | HOST/SYSID |
|---|---|---|---|---|---|---|---|---|---|
| DAB2CAT3 | EXCL | VALID | MONO | 2 | 640 | 32770 | 33409 | | abcdefgh/123 |
| DAB1CAT2 | EXCL | VALID | MONO | 1 | 640 | 2 | 641 | | stuvwxyz/124 |
| DABDCAT1 | SHARED | VALID | DUAL | 1/2 | 640 | 1920 | 2559 | ALLOWED | abcdefgh/123<br>stuvwxyz/124 |
| GSV00001 | SHARED | VALID | DUAL | 1/2 | 1024 | 2560 | 3583 | ALLOWED | abcdefgh/123<br>stuvwxyz/124 |

Diese Anzeige ergibt sich für die im Beispiel konfigurierten Partitions:

- ACCESS zeigt an, ob die Partition genutzt wird und im Ja-Fall, ob exklusiv oder im Shared-Modus.
- DATA zeigt an, ob die Daten in der Partition gültig sind.

- MODE unterscheidet Mono- von Dual-Partitions.
- ATTACH-DUAL zeigt an, ob nach dem Wegschalten einer GS-Unit (nach Fehler oder DETACH-GS-UNIT-Kommando) das Wieder-Hinzuschalten dieser GS-Unit vom Nutzer der Partition erlaubt wird (diese Anzeige ist nur bei Dual-Partitions von Bedeutung).
- GSU gibt die Lage der Partition auf den GS-Units wieder.
- BEGIN und END geben bei Mono-Partitions die Positionen auf der jeweiligen GS-Unit an, bei Dual-Partitions die Positionen auf GS-Unit 1.
- HOST/SYSID liefert die Rechnernamen und/oder die Systemkennung (SYSID) der BS2000-Systeme, die die Partition aktuell nutzen (in dieser Tabelle wurden nur symbolische Namen verwendet).



**ACHTUNG!**
Die Anzeige MODE=DUAL in der Ausgabe von SHOW-GS-STATUS zeigt nicht den aktuellen Arbeitsmodus der Partition, sondern ihren „Soll-Modus“ an.
Der tatsächliche aktuelle Stand ergibt sich daraus unter Berücksichtigung der GS-Unit-Zustände:
Wenn beide GS-Units im Status ATTACHED sind, wird dual aufgezeichnet.
Ist eine GS-Unit im Status DETACHED, wird lediglich mono aufgezeichnet.

*Abschnitt 4: Free-GS-Space-Daten*

Der freie GS-Adressraum wird in zusammenhängenden Mono- und Dual-Stücken angezeigt.

FREE MONO GS-SPACE:

| GS-UNIT | SIZE | BEGIN | END |
|---|---|---|---|
| 1 | 1278 | 642 | 1919 |
| 1 | 512 | 3584 | 4095 |
| 2 | 1278 | 33410 | 34687 |
| 2 | 512 | 36352 | 36863 |

FREE DUAL GS-SPACE:

| GS-UNIT | SIZE | BEGIN | END |
|---|---|---|---|
| 1/2 | 1278 | 642 | 1919 |
| 1/2 | 512 | 3584 | 4095 |

### 4.3.3 GS-Rekonfiguration

Die GS-Rekonfiguration besteht aus dem Zu- und Wegschalten von GS-Units über die angebotenen Hardware-Schnittstellen. Das Wegschalten geschieht implizit bei GS-Unit-Ausfall und explizit mit dem Kommando DETACH-GS-UNIT. Das Zuschalten geschieht implizit bei Systemstart und explizit mit dem Kommando ATTACH-GS-UNIT.
Die hardwareseitig verfügbaren GS-Units müssen also nicht explizit zugeschaltet werden. Das Kommando SHOW-GS-STATUS SELECT=*GS-UNITS zeigt nach dem Systemstart die verfügbaren GS-Units und ihren Status an.

Zur GS-Rekonfiguration im XCS-Verbund siehe Abschnitt „GS-Betrieb im XCS-Verbund“ auf Seite 149.

### Wegschalten einer GS-Unit

Kommt es im laufenden System zu einem GS-Unit-Ausfall, wird die betroffene GS-Unit vom System weggeschaltet (Meldung `EGC2100`).

Das Kommando DETACH-GS-UNIT GS-UNIT=x wird nur dann angenommen, wenn keine der Mono-Partitions auf der betroffenen GS-Unit benutzt wird; im Fall, dass die letzte GS-Unit weggeschaltet wird, darf auch keine Dual-Partition benutzt sein (Anzeige jeweils ACCESS=NONE).
Ggf. müssen noch laufende Nutzungen beendet werden.
Informationen über die Nutzung auf einer GS-Unit erhält man über das Kommando SHOW-GS-STATUS SELECT=*PARTITIONS(SELECT=*BY-ATTRIBUTES(GS-UNIT=x)).

Nach dem Wegschalten der zweiten GS-Unit sind Dual-Partitions auf der verbleibenden GS-Unit weiter zugreifbar. Es muss jedoch beachtet werden, dass sich – nach dem ersten Schreibzugriff auf Daten auf der verbleibenden GS-Unit – auf der weggeschalteten GS-Unit veraltete Daten befinden.

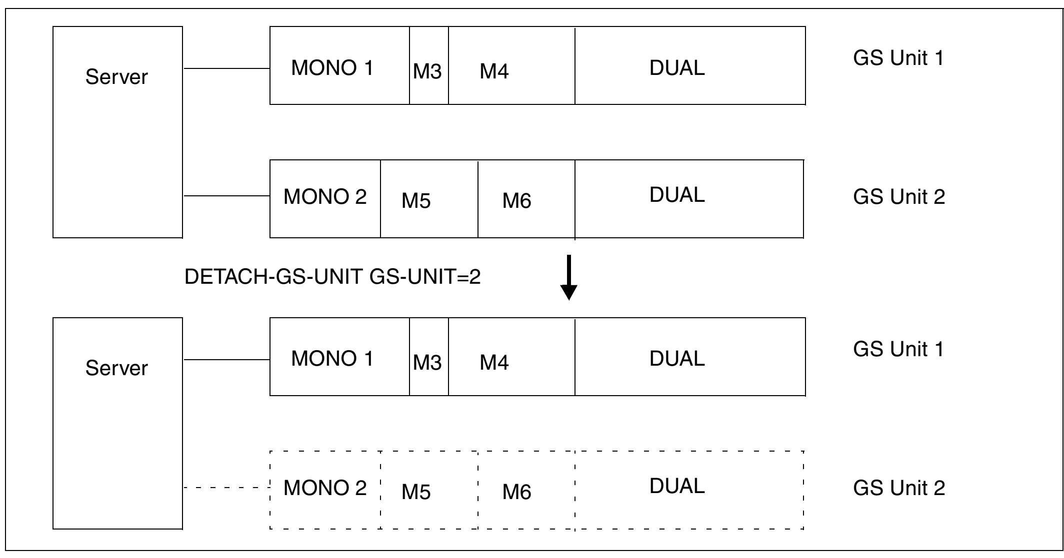



Bild 4: Wegschalten der zweiten GS-Unit mit DETACH-GS-UNIT

### Zuschalten einer GS-Unit

Das Kommando ATTACH-GS-UNIT GS-UNIT=x wird nur dann angenommen, wenn die GS-Unit ONLINE, aber nicht ATTACHED ist (Anzeige STATE=DETACHED).
Beim Zuschalten der zweiten GS-Unit müssen außerdem die aktuellen Nutzer von Dual-Partitions das Zuschalten der zweiten GS-Unit erlauben (Anzeige ATT-DUAL=ALLOWED). Im Konfliktfall müssen ggf. die Nutzungen beendet werden, die das Zuschalten einer zweiten GS-Unit nicht erlauben.
Über solche Nutzungen kann sich der Systembetreuer mit folgendem Kommando informieren: SHOW-GS-STATUS SELECT=*PARTITIONS(SELECT=*BY-ATTR(ATTACH-DUAL= *FORBIDDEN)).

Im Verlauf des ATTACH-GS-UNIT-Kommandos werden automatisch alle Dual-Partitions mit gültigen Daten egalisiert, d.h. die Daten solcher Partitions werden von der verbliebenen GS-Unit auf die zuzuschaltende GS-Unit kopiert. Die Ausführung dieses Kommandos kann daher u.U. mehrere Minuten dauern. Schreibzugriffe auf den GS während der Egalisierung werden von GSMAN bzw. den GS-Nutzern synchronisiert. Nach erfolgreicher Kommandobeendigung sind die Dual-Partitions mit gültigen Daten auf beiden GS-Units identisch.

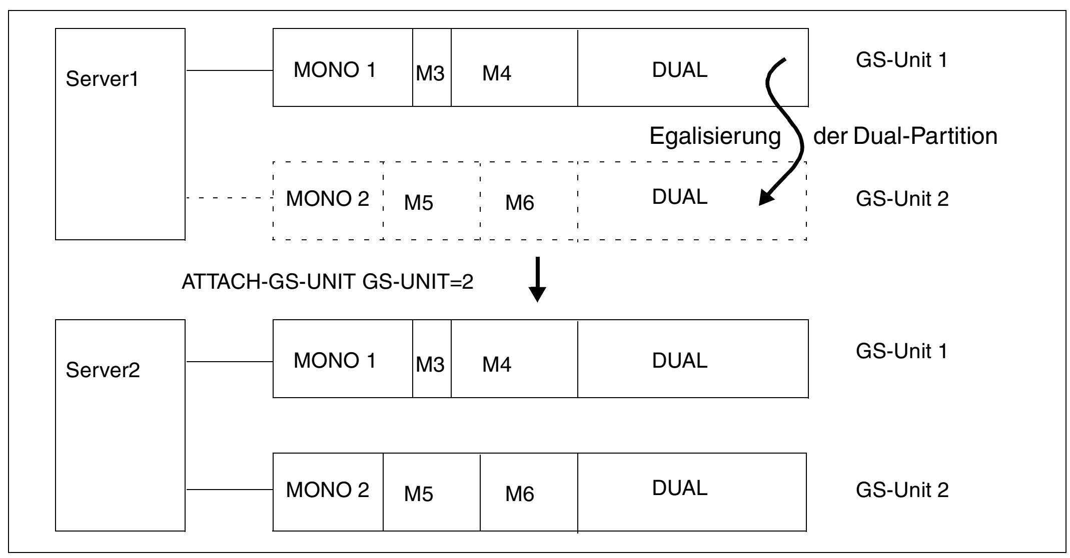



Bild 5: Zuschalten der zweiten GS-Unit mit ATTACH-GS-UNIT

### Besonderheiten auf VM2000

In einem VM2000-System kann Gastsystemen entweder ein virtueller oder ein realer GS zugewiesen werden (siehe Handbuch „VM2000“ [62]).

Die Zuweisung des GS über das VM2000-Kommando ADD-VM-RESOURCES VM-ID=..., GS=*REAL/*VIRTUAL(VIRTGSxx) sollte für alle Gastsysteme vor der Systemeinleitung des Gastsystems (/START-VM) erfolgen. Die Zuweisung kann auch nachträglich erfolgen, siehe dazu Abschnitt „GS-Complex-Konfiguration“ auf Seite 158.

Ein zugewiesener virtueller GS kann bei laufendem Gastsystembetrieb in seiner Größe geändert werden. Dazu sind folgende Schritte nötig:

1. Gastsystem: `DETACH-GS-UNIT GS-UNIT=1`
   `[ggf. DETACH-GS-UNIT GS-UNIT=2]`
2. VM2000: `REMOVE-VM-RESOURCES VM-ID=...,GS=*VIRTUAL(VIRTGSxx)`
3. Monitor: Ändern der Partition-Größe (oder Auswahl einer neuen Partition VIRTGSyy, ggf. mit Kopieren der Daten von VIRTGSxx nach VIRTGSyy)
4. VM2000:
```
ADD-VM-RESOURCES VM-ID=...,GS=*VIRTUAL(VIRTGSxx),
                 VM-SHARED=...
```
5. Gastsystem: `ATTACH-GS-UNIT GS-UNIT=1`
   `[ggf. ATTACH-GS-UNIT GS-UNIT=2]`

Das Gastsystem nutzt danach den GS in seiner neuen Größe.

Alternativ kann im 4. Schritt dem Gastsystem auch eine andere Partition als virtueller GS zugewiesen werden.

### 4.3.4 GS-Betrieb im XCS-Verbund

Bezüglich Globalspeicher ermöglicht XCS (Cross-Coupled-System) die gemeinsame Nutzung eines GS durch alle beteiligten Systeme (Cluster-Elemente) im globalen GS-Betrieb (Shared Global Storage). Dabei wird vorausgesetzt, dass alle Systeme im XCS einen „Parallel HIPLEX“ bilden (siehe Bild 7 auf Seite 151).
Daneben ist im XCS auch ein lokaler GS-Betrieb möglich: einzelne Systeme nutzen einen eigenen GS exklusiv (siehe Bild 6 auf Seite 151).
Mischformen zwischen lokalem und globalem Betrieb (z.B. nutzen zwei Systeme einen GS gemeinsam, ein drittes System hat einen eigenen oder keinen GS) sind nicht möglich.

Die Steuerung des GS-Betriebes erfolgt über den Subsystem-Parameter GS-USAGE, der in der Datei SYSSSI.GSMAN.180 hinterlegt und beim Start des Subsystems GSMAN ausgewertet wird. Die möglichen Werte sind LOCAL und GLOBAL.



Es werden folgende Schreibweisen akzeptiert:
- GSUSAGE=LOCAL und GSUSAGE=GLOBAL
- GS-USAGE=*GLOBAL und GS-USAGE=*LOCAL
- GS-USAGE=GLOBAL und GS-USAGE=LOCAL

Akzeptiert werden auch führende, trennende und nachlaufende Leerzeichen.

Der Wert GS-USAGE=GLOBAL setzt voraus, dass das BS2000-System an einem XCS-Verbund teilnehmen kann. Ist dies nicht gegeben, ändert das Subsystem GSMAN ggf. von sich aus den Wert auf GS-USAGE=LOCAL (unter Ausgabe der Meldung `EGC1012`).

Zum XCS-Verbund siehe Abschnitt „XCS-Verbund“ auf Seite 403.

### Ändern des Parameters GS-USAGE im laufenden Betrieb

Voraussetzung für das Ändern des Wertes des Parameters GS-USAGE im laufenden Betrieb ist, dass:

1. der Globalspeicher aktuell nicht genutzt wird (ACCESS=NONE bei allen Partitions in der Kommandoausgabe von SHOW-GS-STATUS SELECT=*PARTITIONS).
2. das Subsystem MSCF nicht gestartet ist oder der XCS aktuell nur aus dem eigenen System besteht.

Die Änderung erfolgt über das Kommando MODIFY-GSMAN-PARAMETER ..., GS-USAGE=LOCAL/GLOBAL.

Sofern GS-Volumes genutzt werden oder zu einem späteren Zeitpunkt genutzt werden sollen, ist nach einer Änderung des Parameters GS-USAGE das Subsystem GSVOL zu beenden und neu zu starten. Das Beenden von GSVOL erfordert wiederum das Wegschalten (DETACH) von ggf. zugeschalteten GS-Volumes.

Die aktuelle Einstellung des GS-USAGE-Parameters wird mit dem Kommando SHOW-GS-STATUS ausgegeben – unabhängig von der speziellen Operandenauswahl.

### Konfigurationsprüfungen durch GSMAN im XCS-Verbund

Beim Eintritt eines BS2000-Systems in einen XCS-Verbund, der durch den Start des Subsystems MSCF eingeleitet wird, stellt das Subsystem GSMAN sicher, dass alle beteiligten Systeme den gleichen Wert für GS-USAGE eingestellt haben. Dazu bestimmt das erste System mit seiner Einstellung von GS-USAGE den Sollwert im XCS-Verbund; bei nachfolgenden Systemen wird der Eintritt in den XCS-Verbund von GSMAN abgebrochen, wenn der eigene Wert von GS-USAGE nicht mit dem des ersten Systems übereinstimmt (Meldung EGC1007).

Weiterhin überprüft das Subsystem GSMAN, ob die XCS-Konfiguration mit der HW-Konfiguration verträglich ist: Im Falle GS-USAGE=GLOBAL müssen alle Systeme aus Hardware-Sicht Systeme am gleichen GS-Complex sein. Im Konfliktfall bricht GSMAN den Eintritt in den XCS-Verbund ab (Meldung EGC1005).

### Lokaler GS-Betrieb im XCS-Verbund

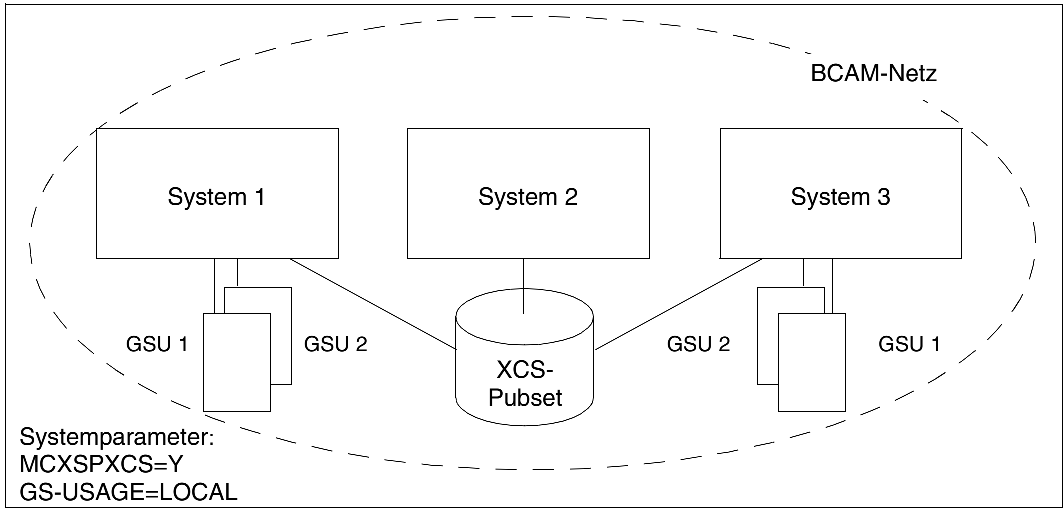



Bild 6: Lokaler GS-Betrieb in einem XCS-Verbund mit 2 Systemen mit eigenem GS und einem System ohne GS

Die Verwaltung und Rekonfiguration von lokalem GS entsprechen den jeweiligen Funktionen auf einem System ohne XCS. Das Gleiche gilt für die GS-Nutzer. Insbesondere wird NSM in diesem Fall den GS nicht zur Kommunikation nutzen.

### Globaler GS-Betrieb im XCS-Verbund: Parallel HIPLEX

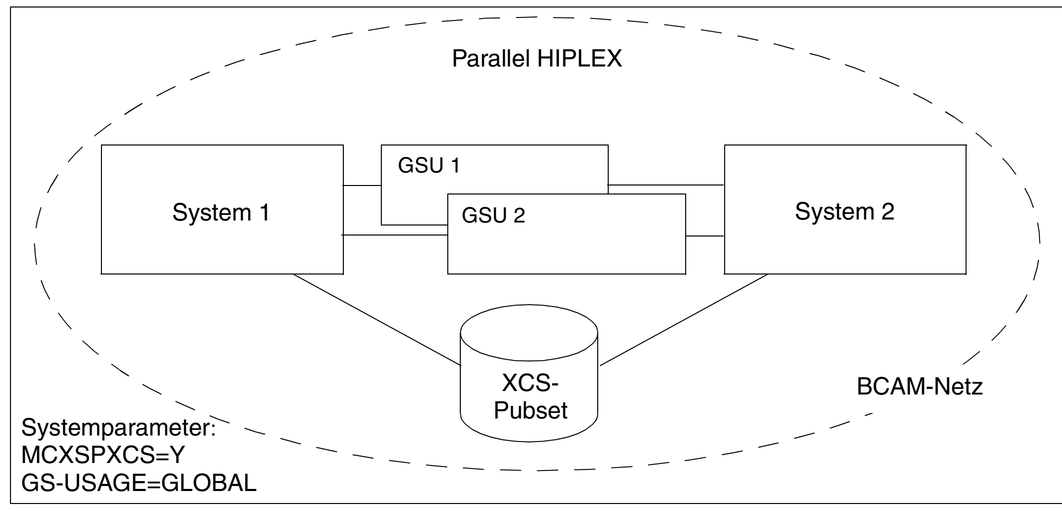



Bild 7: Globaler GS-Betrieb in einem XCS-Verbund mit Parallel HIPLEX, bestehend aus 2 Systemen

- Systemvorbereitung

  Vor der gemeinsamen Nutzung eines GS durch mehrere Systeme in einem Parallel HIPLEX sollte in jedem beteiligten System vor dem Systemstart die Datei SYSSSI.GSMAN.180 so modifiziert werden, dass sie den Satz „GS-USAGE=GLOBAL“ enthält.

- GS-Konfiguration

  Alle Aktionen zur Konfiguration des GS sind an einem System durchführbar (Single Point of Administration) und gelten dann XCS-weit, da zu jeder Zeit die gültigen Konfigurationsdaten auf dem GS selbst stehen. Damit verhalten sich alle GS-Konfigurationskommandos im XCS-Verbund genau so wie im sog. „Stand-alone-Fall“. Nach Stopp oder Ausfall des Subsystems MSCF auf einem System werden jedoch dort alle GS-Konfigurationskommandos zurückgewiesen.

- GS-Rekonfiguration

  Die Aktionen zur Rekonfiguration des GS werden ebenfalls XCS-weit ausgeführt und zwar sowohl im Fehlerfall (impliziter DETACH) als auch die explizit über die Kommandos DETACH-GS-UNIT und ATTACH-GS-UNIT angestoßenen Aktionen. Die Überprüfungen vor der Ausführung der Kommandos finden entsprechend XCS-weit statt.

- Nutzungseinschränkungen im globalen GS-Betrieb

  Im Fall GS-USAGE=GLOBAL stehen die Funktionen des Subsystems GSMAN, d.h. alle GS-Kommandos und alle GS-Zugriffe, erst nach erfolgreichem Eintritt in den XCS-Verbund (nach `SUBSYSTEM MSCF READY`) zur Verfügung.

  Mit Austritt aus dem XCS-Verbund werden alle Funktionen von GSMAN wieder gesperrt bis auf die GS-Zugriffe zu den exklusiven Partitions, die zum Zeitpunkt des Austritts aus dem XCS noch in Benutzung sind. Dies gilt insbesondere (aber nicht nur) für den Fall einer abnormalen XCS-Beendigung auf einem System. Vor einem geplanten XCS-Austritt (STOP-SUBSYSTEM MSCF) sollten möglichst alle GS-Nutzungen auf dem betroffenen System über die nutzerspezifischen Kommandos beendet werden. Dazu bietet GSMAN das Kommando SHOW-GS-STATUS *PARTITION(SELECT= *BY-ATTRIBUTES(ASSIGNED-HOST=<hostname>)) an: Es zeigt die Partitions, die vom spezifizierten System noch genutzt werden. Eine Beendigung der GS-Nutzung wird aber auch im geplanten Fall nicht erzwungen.

- Shared-Partitions

  Die Eigenschaft von Partitions, Zugriffe von mehreren Systemen eines XCS-Verbundes aus zu gestatten, wird bei der ersten Anforderung der Partition durch den Nutzer festgelegt.

  Die Nutzer GSVOL und NSM fordern ihre GS-Partitions in einem XCS-Verbund stets shared an. DAB nutzt seine Partitions exklusiv für Cache-Bereiche von exklusiv importierten Pubsets, kann diese aber, falls nötig, auf andere Systeme „umschalten“. Zum Cachen von Shared-Pubsets nutzt DAB GS-Partitions shared und teilt diese in Slices, eine pro beteiligtem System, ein. GSVOL legt im XCS-Verbund die GS-Volumes prinzipiell als Shared GS-Volumes an, die sowohl exklusiv als auch shared importiert werden können. NSM nutzt seine Shared-Partitions zur schnellen Kommunikation im XCS-Verbund.

*GS-Nutzer und ihre exklusiven und Shared-Partitions im XCS-Verbund*

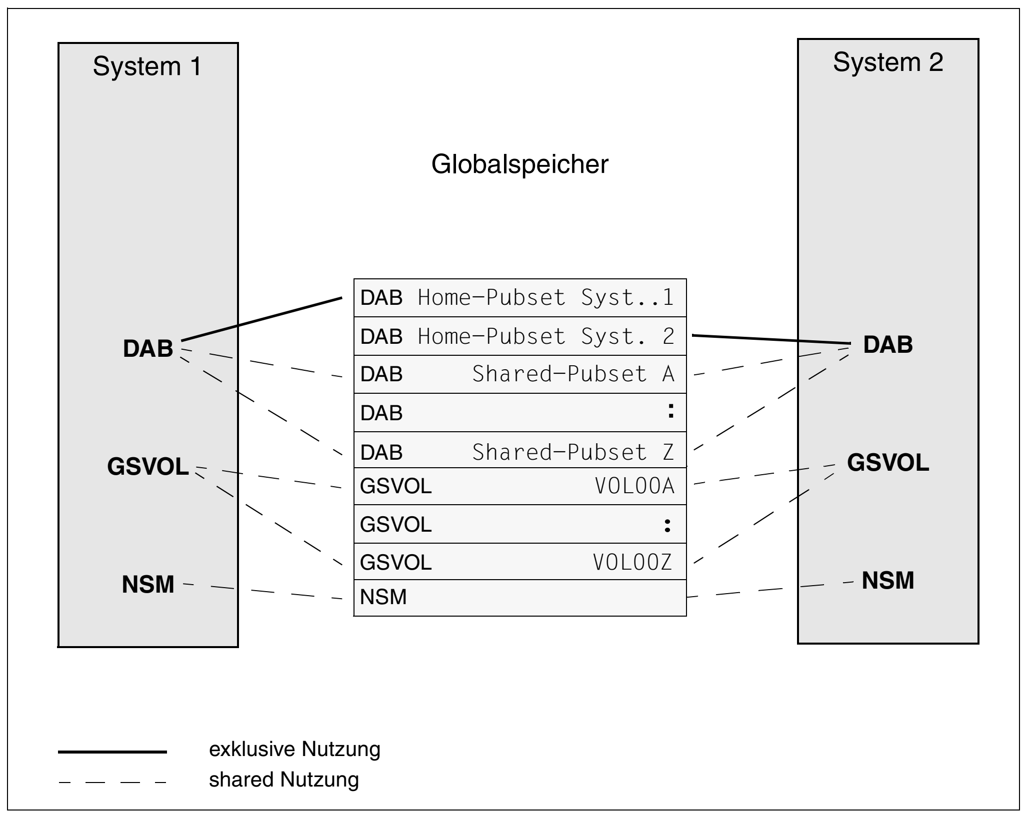



Bild 8: Shared-GS-Nutzung

- Umschalten von exklusiv genutzten Partitions nach abnormaler Systembeendigung oder XCS-Beendigung

  Nach einer abnormalen Systembeendigung eines Systems oder nach XCS-Beendigung auf einem System besteht u.U. die Notwendigkeit, von DAB im GS gecachte Pubsets an einem im XCS verbliebenen System zu importieren. Beim IMCAT auf dem betreffenden System fordert das Subsystem DAB bei GSMAN die zugehörige Partition an. Da aber die fremde Nutzung dieser Partition auf dem GS nicht ausgetragen ist, erscheint auf der Konsole die Meldung:
  `EGC0502 Partition '(&00)' is reserved by system '(&01)'/'(&02)'`

  Die anschließend erscheinende Frage muss mit `Y` beantwortet werden:
  `EGC0503 Do you want to unlock partition '(&00)'? Reply (Y=YES; N=NO)`

  Das Gleiche gilt für sonstige exklusiv genutzte Partitions, wie z.B. VIRTGS-Partitions, die an verbleibenden Systemen im XCS-Verbund genutzt werden sollen.



**ACHTUNG!**
Bei Beantwortung der Frage `EGC0503` mit `Y` muss sicher sein, dass die Nutzung der Partition durch das angezeigte System (&01) in der Meldung `EGC0502` beendet ist!

Das Umschalten von exklusiv genutzten Partitions ist nicht möglich, wenn das nutzende System (HOST/SYSID in der Ausgabe des Kommandos SHOW-GS-STATUS *PARTITION(PARTITION-ID = pppppppp)) sich im XCS-Verbund befindet; in diesem Fall muss erst die Nutzung an diesem System über entsprechende Benutzer-Schnittstellen beendet werden.

### Fehlerbehandlung im Subsystem GSMAN

Das Subsystem GSMAN beinhaltet Funktionen zur Behandlung von Hardware-, Software- und Konfigurationsfehlern. Die Hardware meldet beim Zugriff auf den GS Hardware- und Adressfehler; die Software kann in Fehlersituationen Teilfunktionen beenden oder auch eine abnormale Systembeendigung (SETS) einleiten. Die XCS-Konfiguration kann im Widerspruch zur Hardware- oder Software-Konfiguration des GSMAN stehen. Bei den Software- und Konfigurationsfehlern sind u.U. Maßnahmen durch den Systembetreuer notwendig (siehe unten); Hardwarefehler erfordern in der Regel Servicemaßnahmen.

- Hardware-Fehler

  Bei Hardware-Fehlern auf einer GS-Unit werden Zugriffe auf Mono-Partitions, die auf der betroffenen GS-Unit liegen, mit einem Fehler beendet. Zugriffe auf Dual-Partitions laufen auf der intakten GS-Unit weiter.
  Bei schweren Hardware-Fehlern wird die defekte GS-Unit implizit detached unter Ausgabe folgender Meldung:
  `EGC2100 GSU '(&00)' detached by system`

  Die GS-Nutzer reagieren auf gemeldete Fehler und Warnungen mit eigenen Fehlermeldungen; ob und in welchen Situationen Maßnahmen des Systembetreuers notwendig werden, ist bei den einzelnen Nutzern beschrieben. GS-Hardware-Fehler werden grundsätzlich im Hardware-Error-Logging festgehalten.

- Software-Fehler

  Software-Fehler äußern sich durch den Ausfall von Teilfunktionen oder des gesamten Systems (SETS). Das Subsystem GSMAN behandelt zwei wesentliche SW-Fehler:

  *Abnormale Systembeendigung (SETS)*

  Beim Ausfall eines Systems im XCS-Verbund können alle Daten an verbleibenden Systemen weiter genutzt werden. Exklusiv auf dem ausgefallenen System genutzte Partitions können von anderen Systemen genutzt werden nach Ausgabe der Meldung:
  `EGC0502 Partition '(&00)' is reserved by system '(&01)'/'(&02)'`

  Die anschließend erscheinende Frage muss mit `Y` beantwortet werden:
  `EGC0503 Do you want to unlock partition '(&00)'? Reply (Y=YES; N=NO)`

  Beim Ausfall des letzten Systems im XCS-Verbund bleiben die Daten auf dem GS erhalten und werden nach Wiederanlauf (Systemstart und „`SUBSYSTEM MSCF READY`“) des Systems zur Verfügung gestellt. Exklusiv genutzte Partitions werden auf dem System, der sie zuletzt genutzt hat, ohne Rückfrage zugeteilt, auf anderen Systemen muss die Frage `EGC0503` mit `Y` beantwortet werden.

*Abnormale XCS-Beendigung*

Bei einer abnormalen Beendigung des XCS auf einem System werden dort keine GS-Konfigurations- oder Rekonfigurationskommandos mehr angenommen. Ebenso werden Anforderungen zur Nutzung von Partitions zurückgewiesen. Bestehende Nutzungen können jedoch weiterlaufen. Je nachdem ob der Nutzer die Partition tatsächlich freigibt oder nicht, kann diese auf anderen Systemen genutzt werden nach Ausgabe der Meldung:

```
EGC0502 Partition '(&00)' is reserved by system '(&01)'/'(&02)'
```

Die anschließend erscheinende Frage muss mit `Y` beantwortet werden:

```
EGC0503 Do you want to unlock partition '(&00)'? Reply (Y=YES; N=NO)
```

Sobald nach XCS-Ausfall ein XCS-Restart erfolgt, ist die GS-Nutzung auf diesem System wieder ohne Einschränkungen möglich.

- Konfigurationsfehler

Beim Start des Subsystems GSMAN wird der Subsystemparameter GS-USAGE aus der Datei SYSSSI.GSMAN.180 ermittelt. Erwartet wird ein Satz mit den möglichen Inhalten „GS-USAGE=LOCAL“ oder „GS-USAGE=GLOBAL“.

Bei allen Fehlern (DVS-Fehler oder falsche Satzinhalte) wird GS-USAGE=LOCAL angenommen und mit einer der Meldungen quittiert:

```
EGC1010 DMS ERROR '(&00)' WHEN PROCESSING FILE '(&01)'.
        IN SYSTEM MODE /HELP-MESS DMS(&00). GS-USAGE=LOCAL ASSUMED
EGC1011 Wrong record '(&00)' in subsystem-info-file. GSUSAGE=LOCAL assumed
```

Im Fall, dass das eigene System nicht System in einem XCS-Verbund werden kann, wird der Parameter GS-USAGE=GLOBAL durch GS-USAGE=LOCAL ersetzt und folgende Meldung ausgegeben:

```
EGC1012 Warning: Subsystem-parameter GSUSAGE=GLOBAL ignored.
        GSUSAGE=LOCAL assumed
```

Es wird geprüft, ob ein XCS-Verbund den GS bereits nutzt. Im Konfliktfall stellt GSMAN die Frage:

```
EGC1003 Global Storage is reserved by XCS '(&00)' and usage is global.
        Do you want to change Global Storage usage to local?
        Reply (Y=YES; N=NO)
```

Bei Antwort `Y` muss dahingehend geprüft werden, dass der angegebene XCS-Verbund den GS nicht benutzt. Bei Antwort `N` wird der Systemstart ohne GS-Nutzung fortgesetzt.

Weiter wird geprüft, ob ein System, der nicht in einem XCS ist, den GS nutzt. Im Konfliktfall stellt GSMAN die Frage:

```
EGC1004 Global Storage is reserved by system '(&00)'/'(&01)'.
        Do you want to unlock Global Storage? Reply (Y=YES; N=NO)
```

Bei Antwort `Y` muss dahingehend geprüft werden, dass das angegebene System den GS nicht benutzt. Bei Antwort `N` wird der Systemstart ohne GS-Nutzung fortgesetzt.

Im Auslieferungszustand enthält die Datei den Satz „GS-USAGE=LOCAL“. Vor dem ersten Einsatz eines globalen GS-Betriebs sollte diese Datei editiert werden; das Kommando MODIFY-GSMAN-PARAMETERS ist nur zur Vermeidung von System-Restarts in Fehlersituationen gedacht.

Die Hardware- und Software-Konfiguration im XCS-Verbund wird von GSMAN auf nachfolgende Bedingungen hin beim Start des Subsystems MSCF geprüft:

1. Alle Systeme in einem XCS-Verbund haben den gleichen Wert im Subsystem-Parameter GS-USAGE. Das erste System bestimmt diesen Wert im XCS.
   Der Start des Subsystems MSCF auf einem weiteren System wird vom Subsystem GSMAN verhindert, wenn sein Parameter GS-USAGE einen anderen Wert besitzt.
   Meldung:
   ```
   EGC1007 Error in XCS configuration of Global Storage
   ```

2. Im zweiten Schritt wird weiter geprüft, ob ein anderer XCS-Verbund den GS bereits nutzt. Im Konfliktfall stellt GSMAN die Frage:
   ```
   EGC1002 Global Storage is reserved by XCS '(&00)'.
           Do you want to unlock Global Storage? Reply (Y=YES; N=NO)
   ```
   Es muss geprüft werden, dass der angegebene XCS den GS nicht benutzt (Antwort `Y`). Bei Antwort `N` wird der Start des Subsystems MSCF abgebrochen.

3. Im dritten Schritt wird weiter geprüft, ob ein System, das nicht in einem XCS ist, den GS nutzt. Im Konfliktfall stellt GSMAN die Frage
   ```
   EGC1004 Global Storage is reserved by system '(&00)'/'(&01)'.
           Do you want to unlock Global Storage? Reply (Y=YES; N=NO)
   ```
   Es muss geprüft werden, dass das angegebene System den GS nicht benutzt (Antwort `Y`). Bei Antwort `N` wird der Start des Subsystems MSCF abgebrochen.

4. Im vierten Schritt wird geprüft, dass alle aktuellen Systeme physikalisch mit demselben GS verbunden sind. Wenn nicht, wird die folgende Meldung ausgegeben und der Start des Subsystems MSCF abgebrochen:
   ```
   EGC1005 This system is not connected to Global Storage of XCS '(&01)'
   ```

### 4.3.5 GS-Complex-Konfiguration

**GS-Complex**

Bei der Installation von Globalspeicher stellt der Service im CF-Frame der SVPs die Verbindungen zwischen den GS-Units und den mit diesen verkabelten GS-Servern ein und sichert diese in einem Configurationfile CFR. Bei jedem POWER ON der GS-Server werden entsprechend dieser Voreinstellung die GS-Units ONLINE geschaltet. Im laufenden Betrieb eines GS-Servers können GS-Units auch OFFLINE gehen, z.B. nach einem Stromausfall in der GS-Unit bzw. einer GS-Unit-Abschaltung für Wartungszwecke.

Die miteinander mit Attribut „ONLINE“ verbundenen GS-Server und GS-Units bilden einen **GS-Complex** (Global Storage Coupled Server Complex). Ein GS-Complex besteht also aus einer oder zwei GS-Units und einem bis vier GS-Servern. Die GS-Server sind die Server im GS-Complex; sie zeichnen sich dadurch aus, dass alle GS-Units des GS-Complexes zu ihnen ONLINE sind.

Es werden Kommandos zur Rekonfiguration eines GS-Complexes angeboten, da dieser durch partielle Unterbrechungen in den Stromversorgungen seiner Komponenten auseinander fallen kann. Es wird also nicht nur der im BS2000-Startup vorgefundene GS-Complex unterstützt.
Damit wird eine unterbrechungsfreie Wiederinbetriebnahme der GS-Units auch nach Stromausfällen und anderen Katastrophenszenarien gewährleistet. Darüber hinaus werden dadurch erweiterte Funktionen für die Wartung von GS-Units und GS-Server sowie für Backup-Szenarien zur Verfügung gestellt. Im laufenden Betrieb kann die aktuelle Konfiguration mit Hilfe von BS2000-Kommandos geändert werden. Zum besseren Verständnis der Wirkungsweise der Kommandos wird die Zweistufigkeit des Hardware-Konzepts aus Sicht von BS2000/OSD dargestellt.

- Stufe 1

  Ein GS-Complex ist eine Teilmenge von GS-Servern und GS-Units des gesamten Hardware-Verbundes; diese GS-Server und GS-Units sind ONLINE im GS-Complex.
  Zu jedem GS-Complex gehört wenigstens eine GS-Unit und ein GS-Server.
  Andere GS-Server und/oder die zweite GS-Unit des gleichen Hardware-Verbundes können einen zweiten GS-Complex bilden, in diesem Fall sind diese aus Sicht des ersten GS-Complexes FOREIGN.
  Weitere GS-Server und/oder die zweite GS-Unit können auch zu keinem GS-Complex gehören; sie sind damit OFFLINE aus Sicht des ersten GS-Complexes.

- Stufe 2

  Verbindungen zwischen GS-Units und GS-Servern können nur innerhalb eines GS-Complexes aktiviert werden. Eine Verbindung zwischen GS-Unit und GS-Server ist Voraussetzung für den Datentransfer zwischen beiden.
  Hat eine GS-Unit eine Verbindung zu einem GS-Server, so wird die GS-Unit ATTACHED, der GS-Server CONNECTED genannt.
  Eine GS-Unit, die nicht ATTACHED ist, ist DETACHED; ein GS-Server, der nicht CONNECTED ist, ist DISCONNECTED. Diese Eigenschaften sind GS-Complex-global auf Grund der folgenden Symmetrie-Bedingung im GS-Complex:

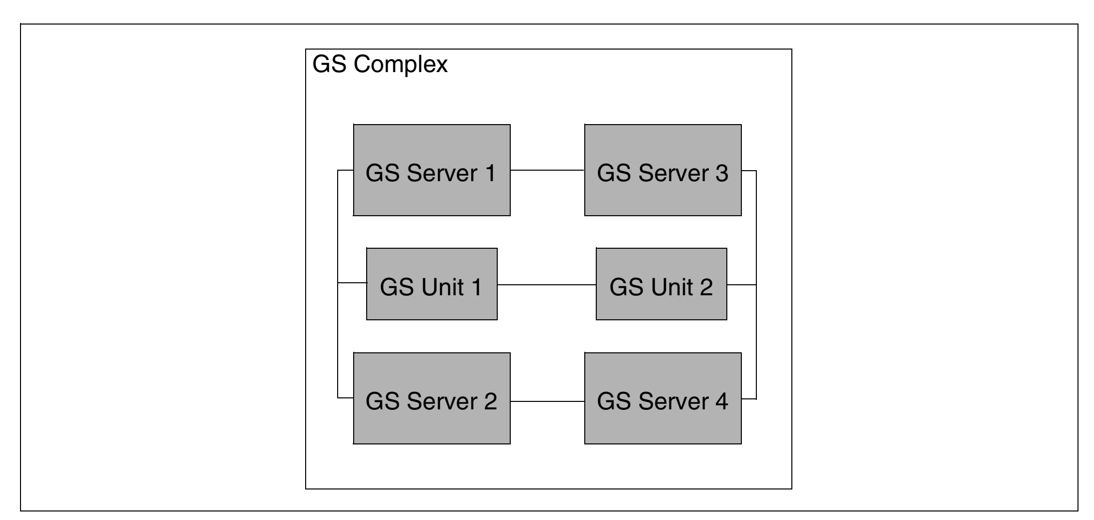
  *Symmetrie-Bedingung im GS-Complex*

  Hat ein GS-Server a eine Verbindung zur GS-Unit x  und ein GS-Server b eine Verbindung zur GS-Unit y (x ungleich y), dann hat auch GS-Server a Verbindung zur GS-Unit y und GS-Server b Verbindung zur GS-Unit x.

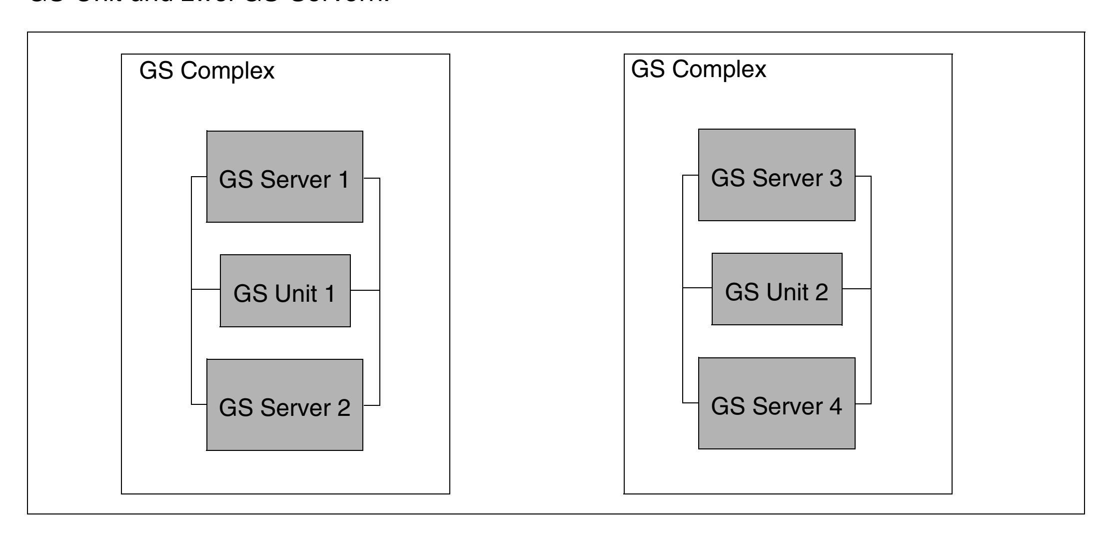
Jeder GS-Server hat eine ID 1 bis16, jede GS-Unit hat eine ID 1 oder 2. Diese IDs dienen der Spezifikation der Objekte in den Kommandos.

*Beispiele*

Auf einer Installation von 4 GS-Servern und 2 GS-Units ist die Grundkonfiguration ein GS-Complex, der aus allen Komponenten besteht. Zur visuellen Darstellung eines GS-Complexes werden die zugehörigen GS-Server und GS-Units durch ein Rechtecknetz geklammert.



Alternativ kann man auf der gleichen Installation zwei GS-Complexe bilden mit je einer GS-Unit und zwei GS-Servern:




### Kommandoübersicht

Auf den o.g. Objekten operieren die folgenden Kommandos:

| **Kommando** | **Bedeutung** |
|---|---|
| *GS-Complex-Rekonfiguration* | |
| CREATE-GS-COMPLEX | GS-Complex bilden |
| MODIFY-GS-COMPLEX | GS-Complex rekonfigurieren |
| SHOW-GS-COMPLEX-CONFIGURATION | Aktuelle GS-Complex-Konfiguration anzeigen |
| *GS-Server-Rekonfiguration* | |
| CONNECT-GS-SERVER | GS-Server mit den GS-Units verbinden |
| DISCONNECT-GS-SERVER | GS-Server von den GS-Units abtrennen |

Tabelle 9: Kommandoübersicht GS-Komplex



**ACHTUNG!**
Diese Kommandos rufen im Hintergrund langlaufende Funktionen der beteiligten SVPs auf. Sie sind daher mit Vorsicht anzuwenden. Andere BS2000-Funktionen, die den SVP benötigen, wie z.B. IOCCOPY, CPU- und Kanal-Rekonfigurationskommandos, sollten nicht gleichzeitig aufgerufen werden.
Die BS2000-Last ist vor der Anwendung dieser Kommandos nach Möglichkeit zu reduzieren.

### GS-Complex mit VM2000

Wenn auf einem GS-Server ein VM2000-System mit mehreren Gastsystemen abläuft, muss der GS zusätzlich mit dem Kommando ADD-VM-RESOURCES zugewiesen werden. Mit dem Kommando REMOVE-VM-RESOURCES wird er wieder entzogen.

Wird mehreren Gastsystemen eines GS-Servers der GS real shared zugewiesen, arbeiten alle Gastsysteme in gleicher Weise auf dem realen GS-Complex.

Für einen virtuellen GS emuliert VMS2000 jeweils einen virtuellen GS-Complex. Für diese stehen die o.g. Kommandos zur GS-Complex-Rekonfiguration nicht zur Verfügung, wohl aber die zur GS-Server- und GS-Unit-Rekonfiguration.

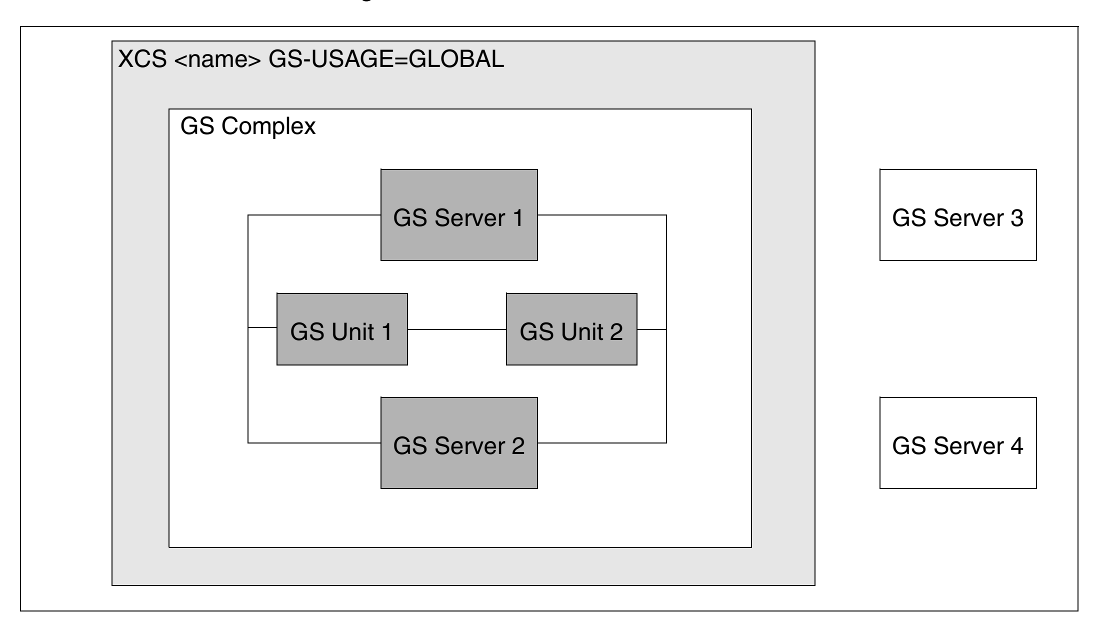
### Informationen über GS-Complexe anzeigen

Die aktuelle GS-Complex-Konfiguration wird mit dem Kommando SHOW-GS-COMPLEX-CONFIGURATION angezeigt. Dabei kann über den Operanden SELECT die Information ausgewählt werden:

- SELECT=*ALL-OBJECTS
  zeigt alle GS-Server und alle GS-Units, die physikalisch miteinander verbunden sind, sowie ihre Zugehörigkeit zu GS-Complexen. Die Information besteht aus den GS-Server-IDs und den GS-Unit-IDs. Der eigene Server (das ist der Kommando-Server) wird gesondert ausgewiesen.
- SELECT=*OBJECTS-OF-OWN-GS-COMPLEX
  zeigt die IDs der GS-Server und GS-Units des eigenen GS-Complexes (das ist der GS-Complex, zu dem der Kommando-Server gehört) an, zusammen mit den zusätzlichen Informationen CONNECT/DISCONNECT der GS-Server und ATTACHED/DETACHED der GS-Units. Soweit auf den konnektierten GS-Servern BS2000-Systeme eines XCS-Verbundes laufen, wird deren VM-ID (falls vorhanden) und Host-Name angezeigt.

*Beispiel*

Eine Installation bestehe aus vier GS-Servern und zwei GS-Units.
Zwei GS-Server (1 und 2) und die beiden GS-Units bilden einen GS-Complex, die beiden anderen Server (3 und 4) seien OFFLINE. Im laufenden Betrieb seien beide GS-Units in Shared-Nutzung im Parallel HIPLEX, der von dem GS-Complex sowie von einem XCS-Verbund, bestehend aus einem native BS2000/OSD auf GS-Server 1 und den Gastsystemen VM2, VM3 auf GS-Server 2, gebildet wird.




Auf GS-Server 1 erhält man dann mit dem Kommando:

/SHOW-GS-COMPLEX-CONFIGURATION  SELECT=*ALL-OBJECTS

```
REAL GS-COMPLEX-CONFIGURATION
OWN GS-SERVER:          1
UNIQUE-GS-COMPLEX:
    GS-SERVERS:         1,2
    GS-UNITS:           1,2
OFFLINE GS-SERVERS:     3,4
OFFLINE GS-UNITS:       -
```

Im Falle eines virtuellen GS-Complexes lautet die Überschrift „VIRTUAL GS-COMPLEX-CONFIGURATION“.

Weitere Information über die Verbindungszustände erhält man durch das Kommando

/SHOW-GS-COMPLEX-CONFIGURATION  SEL=*OBJECTS-OF-OWN-GS-COMPLEX

Es listet alle GS-Units des einen GS-Complexes auf sowie  alle GS-Server des eigenen GS-Complexes, auf denen BS2000-(Gast-)Systeme des eigenen XCS-Verbundes aktiv sind, zusammen mit VM- und XCS-spezifischen Informationen. Im Falle, dass das eigene System den GS lokal nutzt, wird nur der eigene GS-Server mit VM-Information und SYS-ID angezeigt.

```
OWN GS-SERVER:       1

GS-SERVERS OF OWN GS-COMPLEX:

GS-SERVER  STATUS    OPERATION-MODE VM-ID  VM-CONN HOSTNAME  SYSID

1          CONN      NATIVE         -      -       D017ZE01  101

2          CONN      VM2000         2      Y       D017ZE02  102

                     VM2000         3      Y       D017ZE03  103

GS-UNITS OF OWN GS-COMPLEX:

ID         STATUS

1          ATTACHED

2          ATTACHED
```

### 4.3.6 Funktionen und Einschränkungen der Kommandos zur GS-Complex-Rekonfiguration

Einschränkungen in der Anwendbarkeit der Kommandos zur Rekonfiguration eines GS-Complexes ergeben sich aus der oben bereits erwähnten Symmetriebedingung der Hardware, aus der Hardware-Schichtung der Zustände ONLINE/OFFLINE und CONNECT/DISCONNECT sowie aus weiteren Randbedingungen, die sinnvollerweise durch die Software verlangt werden. Letztere sind darin begründet, dass man in einem Parallel HIPLEX den Globalspeicher von allen Knoten aus in gleicher Weise mit den gleichen Attributen sieht.

Daraus leiten sich die Einschränkungen in den nachfolgenden Funktionsbeschreibungen ab:

- CONNECT-GS-SERVER

  Das Kommando CONNECT-GS-SERVER setzt voraus, dass sich der Kommando-Server in einem GS-Complex befindet.
  Es konnektiert den GS-Server mit allen GS-Units des GS-Complexes, falls noch keine GS-Unit im Zustand ATTACHED ist, andernfalls konnektiert es den Kommando-Server an alle GS-Units des GS-Complexes, die bereits im Zustand ATTACHED sind.
  In einem Parallel HIPLEX ist das Kommando nur erlaubt, wenn der Kommando-Server der einzige aktive Knoten im XCS-Verbund ist.

- DISCONNECT-GS-SERVER

  Das Kommando DISCONNECT-GS-SERVER setzt voraus, dass sich der Kommando-Server in einem GS-Complex befindet und dass der GS auf diesem Server nicht mehr genutzt wird.
  Es diskonnektiert den GS-Server von allen GS-Units des GS-Complexes.
  In einem Parallel HIPLEX ist das Kommando nur erlaubt, wenn der Kommando-Server der einzige aktive Knoten im XCS-Verbund ist.

- CREATE-GS-COMPLEX

  Das Kommando CREATE-GS-COMPLEX GS-UNIT=x setzt voraus, dass sich der Kommando-Server und die GS-Unit x nicht in einem GS-Complex befinden.
  Es kreiert einen neuen GS-Complex:
  `EGC3000 Command /(&00) processed with GSU '(&01)'`

- MODIFY-GS-COMPLEX

  1. MODIFY-GS-COMPLEX ACTION=*ADD-GS-UNIT(GS-UNIT=x) setzt voraus, dass sich der Kommando-Server in einem GS-Complex befindet und dass die GS-Unit x OFFLINE ist.
     Es schaltet die GS-Unit x ONLINE/DETACHED.

  2. MODIFY-GS-COMPLEX ACTION=*ADD-GS-SERVER(GS-SERVER=x) setzt voraus, dass sich der Kommando-Server in einem GS-Complex befindet und dass der GS-Server x OFFLINE ist.
     Es schaltet den GS-Server x ONLINE/DISCONNECTED.

     Die Variante MODIFY-GS-COMPLEX ACTION=*ADD-GS-SERVER(GS-SERVER=*OWN(GS-COMPLEX=x)) setzt voraus, dass sich der Kommando-Server in keinem GS-Complex befindet und dass der GS-Complex x existiert.
     Es schaltet den GS-Server ONLINE/DISCONNECTED im GS-Complex x.

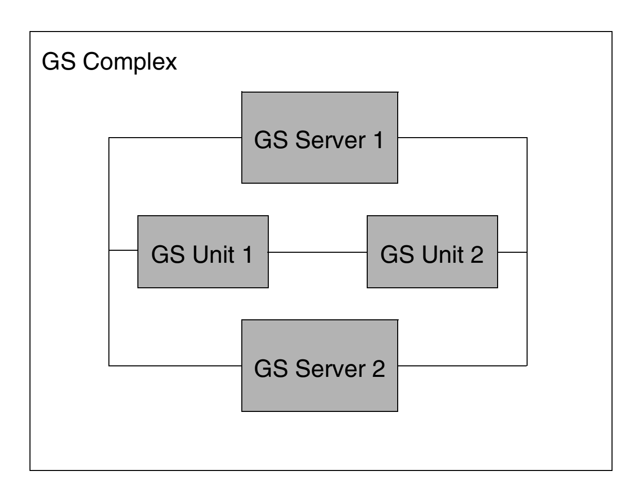
  3. MODIFY-GS-COMPLEX ACTION=*REMOVE-GS-UNIT(GS-UNIT=x) setzt voraus, dass sich der Kommando-Server und die GS-Unit x im gleichen GS-Complex befinden; die GS-Unit x muss im Zustand DETACHED sein.
     Es schaltet die GS-Unit x OFFLINE.

  4. MODIFY-GS-COMPLEX ACTION=*REMOVE-GS-SERVER(GS-SERVER=x) setzt voraus, dass sich der Kommando-Server und der GS-Server x im gleichen GS-Complex befinden; der GS-Server x muss im Zustand DISCONNECTED sein. Der GS-Server x kann auch der Kommando-Server sein.
     Es schaltet den GS-Server x OFFLINE.

### 4.3.7 Beispiele zur GS-Complex-Rekonfiguration

Der Übersichtlichkeit wegen wird im Folgenden eine Konfiguration von zwei GS-Servern und zwei GS-Units betrachtet. Wie bereits erwähnt, beschränkt sich die Anwendung der Kommandos zur GS-Complex-Rekonfiguration auf die unterbrechungsfreie Wiederinbetriebnahme von GS-Units nach Stromausfall und nach dringend notwendigen Wartungsarbeiten sowie auf Backup-Szenarien.

Die Ausgangs- und Zielkonfiguration für die beiden ersten Einsatzfälle ist der folgende GS-Complex:




Die nachfolgenden Ausführungen gelten für Server mit SSU1 bis SSU3; bei der SSU4 wird im POWER-ON einer SSU diese gemäß CFR automatisch ONLINE geschaltet.

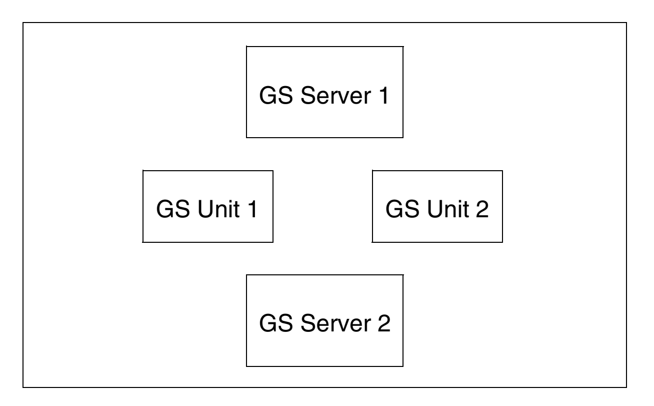
### Stromausfall in beiden GS-Units

Beim Ausfall der Stromversorgung in den GS-Units wird der GS-Complex von den SVPs der GS-Server aufgelöst. Abhängig vom Modell der GS-Units bleibt der GS-Complex auch nach dem Wiedereinschalten des Stroms in den GS-Units aufgelöst.

(auf GS-Server 1:)

```
/SHOW-GS-COMPLEX-CONFIGURATION
```

```
REAL GS-COMPLEX-CONFIGURATION
OWN GS-SERVER:          1
THERE IS NO GS-COMPLEX IN
CONFIGURATION
OFFLINE GS-SERVERS:     1,2
OFFLINE GS-UNITS:       1,2
```

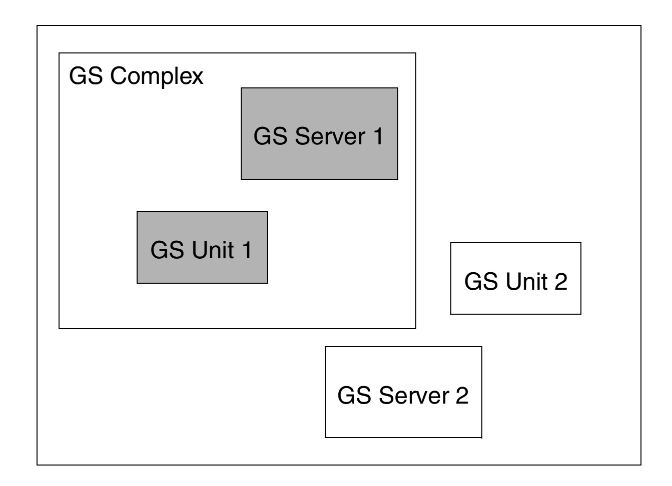



Mit Hilfe der Kommanos CREATE-GS-COMPLEX und MODIFY-GS-COMPLEX kann z.B. von GS-Server 1 aus der Ausgangszustand wie folgt wieder hergestellt werden:

*1. Schritt:*

```
/CREATE-GS-COMPLEX GS-UNIT=1
/SHOW-GS-COMPLEX-CONFIGURATION
```

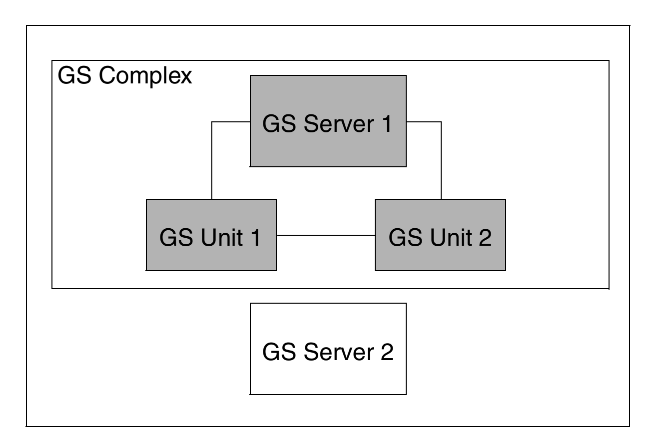
```
REAL GS-COMPLEX-CONFIGURATION
OWN GS-SERVER:            1
UNIQUE-GS-COMPLEX:
    GS-SERVER-ID'S:       1
    GS-UNITS:             1
OFFLINE GS-SERVERS:       2
OFFLINE GS-UNITS:         2
```



*2. Schritt:*

```
/MODIFY-GS-COMPLEX ACTION=*ADD-GS-UNIT(GS-UNIT=2)
/SHOW-GS-COMPLEX-CONFIGURATION
```

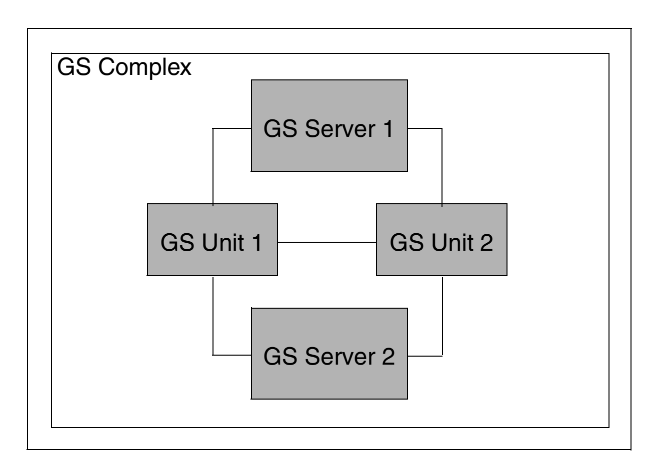
```
REAL GS-COMPLEX-CONFIGURATION
OWN GS-SERVER:          1
UNIQUE-GS-COMPLEX:
    GS-SERVERS:         1
    GS-UNITS:           1,2
OFFLINE GS-SERVERS:     2
OFFLINE GS-UNITS:       -
```




*3. Schritt:*

```
/MODIFY-GS-COMPLEX ACTION=*ADD-GS-SERVER(GS-SERVER=2)
/SHOW-GS-COMPLEX-CONFIGURATION
```

```
REAL GS-COMPLEX-CONFIGURATION
OWN GS-SERVER:          1
UNIQUE-GS-COMPLEX:
    GS-SERVERS:         1,2
    GS-UNITS:           1,2
OFFLINE GS-SERVERS:     -
OFFLINE GS-UNITS:       -
```




*4. Schritt:*

Danach können die GS-Units wieder in Betrieb genommen werden, wobei beachtet werden muss, dass die zuletzt (logisch) weggeschaltete GS-Unit zuerst zugeschaltet werden muss:

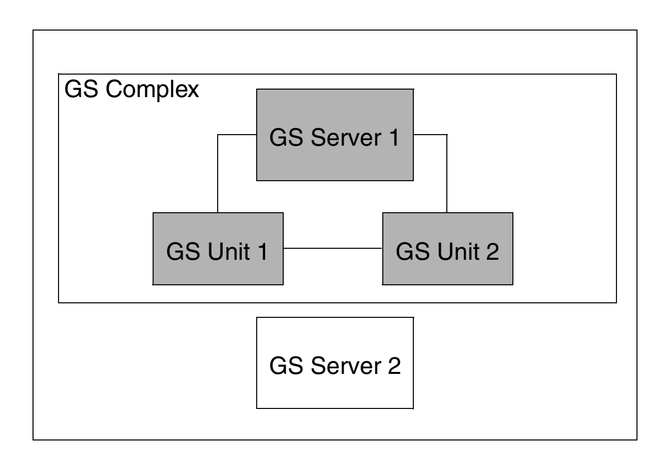
```
/ATTACH-GS-UNIT GS-UNIT=1
/ATTACH-GS-UNIT GS-UNIT=2
```

### Hot Standby Backup-System

In diesem Szenario arbeitet Server 1 mit beiden GS-Units produktiv, Server 2 ist OFFLINE und dient als Testsystem oder für sonstige Aufgaben, die ohne GS auskommen. Bei einem Systemausfall auf Server 1 soll Server 2 die Anwendungen von Server 1 übernehmen, einschließlich der Daten auf dem GS. Die Umschaltung kann z.B durch das Produkt HIPLEX-AF automatisch erfolgen.

```
/SHOW-GS-COMPLEX-CONFIGURATION
```

```
REAL GS-COMPLEX-CONFIGURATION
OWN GS-SERVER:          1
UNIQUE-GS-COMPLEX:
    GS-SERVERS:         1
    GS-UNITS:           1,2
OFFLINE GS-SERVERS:     2
OFFLINE GS-UNITS:       -
```




Vorgehen zum Umschalten auf GS-Server 2:

*1. Schritt:*

Ggf. (Test-)Anwendungen auf Server 2 beenden:

```
/MODIFY-GS-COMPLEX-CONFIGURATION
     ACTION=*ADD-GS-SERVER(GS-SERVER=*OWN(*UNIQUE-GS-COMPLEX))
```

*2. Schritt:*

Solange GS-Server 1 noch mit den GS-Units konnektiert ist, wird das Kommando ATTACH-GS-UNIT zurückgewiesen, wegen der Symmetriebedingungen in einem GS-Complex.

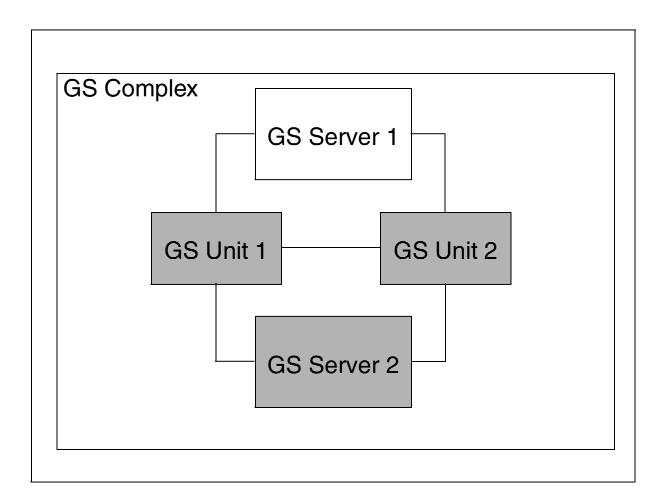
Daher benötigt man in dieser Situation folgendes Kommado, das den GS-Server mit beiden GS-Units konnektiert: /CONNECT-GS-SERVER (auf GS-Server 2),

*3. Schritt:*

Anschließend kann man auf GS-Server 2 die Produktionsanwendungen starten.

```
/SHOW-GS-COMPLEX-CONFIGURATION


REAL GS-COMPLEX-CONFIGURATION
OWN GS-SERVER:            2
UNIQUE-GS-COMPLEX:
    GS-SERVERS:           1,2
    GS-UNITS:             1,2
OFFLINE GS-SERVERS:       -
OFFLINE GS-UNITS:         -
```




GS-Server 1 ist noch in der HW-Konfiguration enthalten, aber software-mäßig ausgefallen. Das Kommando SHOW-GS-COMPLEX-CONFIGURATION wird auf GS-Server 2 ausgeführt.

### 4.3.8 GS-Volumes

GS-Volumes sind emulierte Volumes, deren Daten nur auf dem GS liegen. Für den Nutzer sehen sie weitgehend wie normale Volumes aus, mit der Einschränkung, dass sie nicht für ein Home- oder Paging-Pubset verwendet werden können und dass keine Mischung von GS-Volumes mit normalen Plattenvolumes innerhalb eines Volume-Sets möglich ist. Im Falle eines XCS-Verbunds mit Shared-GS kann auf die GS-Volumes von allen Systemen des XCS-Verbunds aus zugegriffen werden.
Für GS-Volumes müssen die Mnemonics 0300 bis 03FF verwendet werden. Diese Mnemonics werden von IOGEN selbstständig generiert.



Es wird empfohlen, GS-Volumes nur auf einem nicht-flüchtigen GS (stromausfallsicher durch Batterie, Backup oder unterbrechungsfreie Stromversorgung) anzulegen, da andernfalls die Verfügbarkeit einer solchen Platte geringer wäre als die normaler Platten.

| Kommando | Bedeutung |
|---|---|
| CREATE-GS-VOLUME | GS-Volume in GS-Partition einrichten |
| DELETE-GS-VOLUME | GS-Volume löschen |
| SHOW-GS-VOLUME-ATTRIBUTES | Informationen über alle verfügbaren GS-Volumes einholen |

Tabelle 10: Kommandoübersicht für GS-Volumes

#### Anlegen eines GS-Volumes

Zunächst muss eine Partition in der benötigten Größe angelegt werden (siehe Seite 140), deren Name mit „GSV“ beginnen muss.
Dann wird in dieser Partition mit dem Kommando CREATE-GS-VOLUME ein GS-Volume eingerichtet, wobei die zugehörige MN und das Format (welches nicht später mittels VOLIN geändert werden kann) des GS-Volumes festgelegt wird. Der entsprechende GS-Speicherbereich wird durch GSVOL gelöscht.
Die GS-Partition kann im Mono- oder Dual-Modus eingerichtet werden.
Im Dual-Modus wird dabei eine doppelte Aufzeichnung auf den zwei ausfallunabhängigen GS-Units durchgeführt. Diese doppelte Aufzeichnung bei GS-Volumes entspricht den Spiegelplatten des DRV (Dual Recording By Volume). Für GS-Volumes ist entsprechend auch eine Einrichtung als DRV-Platte nicht möglich. Es sollte auch nicht an zwei GS-Volumes die gleiche VSN vergeben werden.



Im VM2000-Betrieb müssen die GS-Volume-MNs den einzelnen Gastsystemen nicht explizit zugewiesen werden.

### Löschen eines GS-Volumes

Mit dem Kommando DELETE-GS-VOLUME kann ein GS-Volume gelöscht werden. Es wird dabei vorausgesetzt, dass das Volume zu keinem System mehr zugeschaltet ist. Andernfalls ist vorher auf allen betroffenen Systemen ein entsprechendes DETACH-DEVICE-Kommando abzusetzen oder der FORCE-Operand zu verwenden.

Wenn vor Systembeendigung die GS-Volumes nicht detached wurden, bleibt die Anzeige, dass die GS-Volumes zu diesem System zugeschaltet sind, stehen.
Diese Situation kann bereinigt werden, indem nach dem erneuten Hochfahren des Systems ein ATTACH-DEVICE- und nachfolgend ein DETACH-DEVICE-Kommando für diese Volumes abgesetzt wird.

### Beispiel

```
/CREATE-GS-PARTITION PARTITION-ID=GSV00001,SIZE=128,
         MODE=*MONO(GS-UNIT=1) ------------------------------------------   (1)
/CREATE-GS-PARTITION PARTITION-ID=GSV00002,SIZE=256,MODE=*DUAL ---------   (2)
/CREATE-GS-VOLUME PARTITION-ID=GSV00001,DEVICE-UNIT=0301,FORMAT=*K -----   (3)
/CREATE-GS-VOLUME PARTITION-ID=GSV00002,DEVICE-UNIT=0302,FORMAT=*NK4 --   (4)
/ATTACH-DEVICE UNIT=0301 -----------------------------------------------   (5)
/ATTACH-DEVICE UNIT=0302 -----------------------------------------------   (6)
/START-SIR
//DECLARE-PUBSET PUBSET-TYPE=*SING-F(PUBSET=GS01,
                 ACTION=*INST(FORMAT=*K),SYS-ID=*STD) ------------------   (7)
//CREATE-VOLUME 0(UNIT=0301) -------------------------------------------   (8)
//CREATE-CATALOG DISK-NUMBER=0,SIZE=300
//START
//DECLARE-PUBSET PUBSET-TYPE=*SING-F(PUBSET=GS02, ----------------------   (9)
                 ACTION=*INST(FORMAT=*NK(PHYS-BLOCK-SIZE=*4K)),SYS-ID=*STD)
//CREATE-VOLUME 0(UNIT=0302) -------------------------------------------  (10)
//CREATE-CATALOG DISK-NUMBER=0,SIZE=300
//START
//END
```

(1) Die GS-Partition mit der Partition-ID `GSV00001` der Größe 128 MB wird in der GS-Unit 1 im Mono-Modus angelegt. Sie ist in der Folge ein Behälter für ein GS-Volume mit einfacher Aufzeichnung.

(2) Die GS-Partition mit der Partition-ID `GSV00002` der Größe 256 MB wird (je 256 MB in GS-Unit 1 und in GS-Unit 2) im Dual-Modus angelegt. Sie ist in der Folge ein Behälter für ein GS-Volume mit doppelter Aufzeichnung.

(3) Die GS-Partition mit der Partition-ID `GSV00001` wird als ein GS-Volume mit der Geräte-Mnemonic `0301` eingerichtet.

(4) Die GS-Partition mit der Partition-ID `GSV00002` wird als ein GS-Volume mit der Geräte-Mnemonic `0302` eingerichtet.

(5) Das GS-Volume mit der Geräte-Mnemonic `0301` wird zugeschaltet.

(6) Das GS-Volume mit der Geräte-Mnemonic `0302` wird zugeschaltet.

(7) Ein SF-Pubset mit der Catid `GS01` (Format K) wird eingerichtet.

(8) Das GS-Volume (Geräte-Mnemonic `0301`) wird als Volume des Pubsets initialisiert.

(9) Ein SF-Pubset mit der Catid `GS02` (Format NK4) wird eingerichtet.

(10) Das GS-Volume (Geräte-Mnemonic `0302`) wird als Volume des Pubsets initialisiert.

**Anzeigen der Attribute der vorhandenen GS-Volumes**

Die Attribute der vorhandenen GS-Volumes können mit folgendem Kommando angezeigt werden:

```
SHOW-GS-VOLUME-ATTRIBUTES *BY-DEVICE-UNIT (DEVICE-UNIT=(0301,0302)),
                          INFORMATION=*ALL.
```

Die Ausgabe ist wie folgt strukturiert:

```
MN       = 0301       VSN         = GS01.0      PARTITION-ID  = GSV00001
GS-UNIT  = 1          SIZE (PHP)  = 65022       FORMAT        = 2K
DUAL     = NO         STATUS      = ACCESS
HOST     = LOCAL

MN       = 0302       VSN         = GS02.0      PARTITION-ID  = GSV00002
GS-UNIT  = 1/2        SIZE (PHP)  = 131070      FORMAT        = NK4
DUAL     = YES        STATUS      = DUAL
HOST     = LOCAL
```

MN gibt die Geräte-Mnemonic des GS-Volumes an, die bei einem XCS-Verbund verbundweit eindeutig ist.

VSN wird dem GS-Volume durch VOLIN oder SIR zugewiesen und muss bei einem XCS-Verbund ebenfalls verbundweit eindeutig sein.

PARTITION-ID
ist die ID der GS-Partition, in der das GS-Volume eingerichtet worden ist.

GS-UNIT
gibt die GS-Unit an, auf der das GS-Volume liegt bzw. 1/2, wenn das GS-Volume in einer Dual-Partition liegt.

SIZE ist die Größe des GS-Volumes in PHP.

FORMAT
ist das Format (K, NK2 oder NK4) des Volumes.

DUAL gibt an, ob das Volume auf einer Mono- oder Dual-Partition liegt.

STATUS
: gibt an, ob das GS-Volume überhaupt zugreifbar ist, und bei DUAL=YES wird angezeigt, ob eine GS-Unit ausgefallen ist.
Mögliche Werte: ACCESS, DUAL, MONO-1, MONO-2, NOACC
(bei MONO-1 ist GS-Unit 2 ausgefallen, bei NOACC beide GS-Units, ...)

HOST
: zeigt die Systeme an, die mit dem GS-Volume verbunden sind, d.h. von denen aus ein ATTACH-DEVICE-Kommando für dieses GS-Volume abgesetzt worden ist.
Wenn der GS ohne XCS oder lokal innerhalb eines XCS-Verbunds genutzt wird, wird unter HOST „LOCAL“ angezeigt.
Die Information HOST erscheint nicht bei INFORMATION=*STD.

## 4.4 Seitenwechselbereich (Paging-Area)

### Paging-Dateien und Seitenwechselbereich

Für die Bereitstellung von virtuellem Adressraum wird Seitenwechselspeicher benötigt, der mit Seitenwechseldateien (Paging-Dateien) realisiert wird. Paging-Dateien tragen immer den Namen `SYS.PAGING.<vsn>`.
Um zum Seitenwechsel herangezogen zu werden, muss eine eingerichtete Paging-Datei explizit dem Seitenwechselspeicher zugewiesen werden.

Die Gesamtheit der dem Seitenwechselspeicher zugewiesenen Paging-Dateien im System heißt Seitenwechselbereich (Paging-Area).

Der Seitenwechselbereich kann während des Systemlaufs dynamisch erweitert oder reduziert werden, indem Paging-Dateien zum Seitenwechselbereich zugewiesen oder aus ihm freigegeben werden. Während der Systemeinleitung ist eine bestimmte Größe und Beschaffenheit des Seitenwechselbereichs Voraussetzung (siehe Seite 181).

Durch die Größe des Seitenwechselbereichs wird der verfügbare virtuelle Adressraum eines Systemlaufs bestimmt, da für jede virtuelle Seite ein Block (zwei 2K-PAM-Seiten) im Seitenwechselbereich vorhanden sein muss. Die max. Gesamtgröße des Seitenwechselbereichs beträgt 4 TB.

Mit Hilfe des Parameterservices können während der Systemeinleitung aus der Menge der eingerichteten und verfügbaren Paging-Dateien diejenigen Dateien ausgewählt werden, die im anschließenden Systemlauf verwendet werden sollen (siehe Abschnitt „Auswahl des Seitenwechselbereichs beim Startup (PAGING)“ auf Seite 115).
Deshalb ist zwischen dem Einrichten von Paging-Dateien (inklusive der Reservierung von Speicherplatz) und der Auswahl von Paging-Dateien für den Systemlauf zu unterscheiden.

Randbedingungen für eine Paging-Datei:

- Sie darf keine Datei ≥ 32 GB sein
- Sie darf auf keinem im GS emulierten Volume stehen
- Sie darf auf keiner Privatplatte eingerichtet werden
- Sie darf auf keinem Shared-Pubset eingerichtet werden

### Paging-Platte und Paging-Pubset

Eine Platte wird als Paging-Platte bezeichnet, wenn dort eine Paging-Datei nicht nur eingerichtet, sondern auch dem Seitenwechselspeicher zugewiesen worden ist.

Generell sind alle Plattentypen erlaubt. Es wird jedoch empfohlen, nur solche Platten zu verwenden, die eine I/O-Rate größer 20 Ein-/Ausgaben pro Sekunde erlauben. Die Platten des Seitenwechselbereichs müssen nicht vom selben Typ sein.

Eine Paging-Datei darf auf einer Platte liegen, die dynamisch als DRV-Volume eingerichtet wurde. (Paging-Dateien, die bei der Systemeinleitung in der Parameterdatei – Parametersatz PAGING – angegeben werden und nicht im Home-Pubset liegen, werden nicht von DRV unterstützt.) Es wird jedoch nur eine der beiden Platten beschrieben.

Ein Pubset wird als Paging-Pubset bezeichnet, wenn mindestens eine Platte des Pubsets als Paging-Platte genutzt wird.

Enthält ein Pubset nur Paging-Dateien, ist er ein reiner Paging-Pubset.
Liegt kein reiner Paging-Pubset vor und besteht der Pubset aus mehreren Platten, sollte keine Paging-Datei auf der Pubres des Pubsets angelegt werden.

Es wird empfohlen, den Seitenwechselbereich überwiegend mit Ein-Platten-Pubsets zu realisieren, da dadurch die Rekonfiguration des Seitenwechselbereichs im laufenden Betrieb bei Aufnahme einer Paging-Datei vereinfacht wird.

| Kommando | Bedeutung |
|---|---|
| CREATE-PAGING-FILE | Paging-Datei erzeugen |
| DELETE-PAGING-FILE | Paging-Datei zerstören (überschreiben mit binär Null) |
| EXTEND-PAGING-AREA | Seitenwechselbereich dynamisch erweitern |
| MODIFY-PAGING-AREA-ATTRIBUTES | Attribute von Paging-Dateien auf Platte bzw. im Globalspeicher ändern |
| REDUCE-PAGING-AREA | Seitenwechselbereich reduzieren |
| SHOW-PAGING-CONFIGURATION | Informationen über genutzte und ungenutzte Paging-Dateien ausgeben |

Tabelle 11: Kommandoübersicht Seitenwechselbereich

Das Kommando SHOW-PAGING-CONFIGURATION gibt folgende Informationen aus:

- Welche Paging-Dateien werden genutzt?
- Welche Paging-Dateien (auf importierten Pubsets) sind ungenutzt?
- Enthält ein Volume eine genutzte oder eine ungenutzte Paging-Datei?
- Welche Gesamtgröße haben die gefundenen genutzen/ungenutzten Paging-Dateien?

### Beispiele für die Ausgabe von Paging-Dateien

```
/SHOW-PAGING-CONFIGURATION VOLUME=*UNUSED,INFORMATION=*ALL
% LIST OF THE FOUND AND UNUSED PAGING-FILES :
%
%  VOLUME  CAT-ID       SIZE                   RESTRICTION
%  1OSH.0   1OSH       256.00 MB                  NONE
%  1OSH.2   1OSH       256.00 MB                  NONE
%  -----------------------------------------------------------------------
%  SUMMARY             512.00 MB

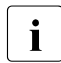
/SHOW-PAGING-CONFIGURATION VOLUME=*USED,INFORMATION=*ALL
% LIST OF THE USED PAGING-FILES :
%
%  VOLUME  CAT-ID       SIZE       FREESIZE     UTILIZATION  REDUCTION  R
%  1OP1.1 * 1OP1      2500.00 MB    1510.99 MB    *LOW       *NOT-REQ   N
%  1OP1.0 * 1OP1      2500.00 MB    2200.00 MB    *LOW       *NOT-REQ   N
%  1OPP.0 * 1OPP      2500.00 MB     395.90 MB    *LOW       *NOT-REQ   N
%  1OSH.1   1OSH       256.00 MB     200.00 MB    *LOW       *NOT-REQ   N
%  -----------------------------------------------------------------------
%  SUMMARY            7756.01 MB
%
%  EMM2317 THE SIZE OF VOLUME(S) - MARKED WITH * - IS INCREASED BY 25%,
            BECAUSE THE PAM-KEYS ARE USED AS PAGING-AREA
```

## 4.4.1 Einrichten, Zuweisen, Freigabe und Löschen von Paging-Dateien

### Einrichten von Paging-Dateien

Paging-Dateien werden unter der Benutzerkennung der Systemverwaltung mit dem Namen `SYS.PAGING.<vsn>` katalogisiert. Eine Paging-Datei darf auf dem Datenträger mit dem angegebenen Datenträgerkennzeichen vsn mehrere Extents haben. In einem Pubset kann je Datenträger genau eine Paging-Datei `SYS.PAGING.<vsn>` existieren.

Eine Paging-Datei hat eine Mindestgröße von 1 MB. Aus Performancegründen wird empfohlen, den Seitenwechselbereich auf mehrere Paging-Dateien (also auf mehrere Datenträger) zu verteilen und die Größe der Paging-Dateien nahezu gleichzuhalten.



Je höher die Paging-Rate ist, desto wichtiger ist die Verteilung des Seitenwechselbereichs auf mehrere Paging-Dateien. Die Paging-Ein-/Ausgaben können beschleunigt werden, indem die verwendeten Platten an unterschiedliche Kanäle angeschlossen werden. Die Paging-Rate kann mit openSM2 (siehe Handbuch „openSM2“ [51]) gemessen werden.

Paging-Dateien können im laufenden Betrieb mit dem Kommando CREATE-PAGING-FILE eingerichtet werden. Durch dieses Kommando wird eine Paging-Datei `SYS.PAGING.<vsn>` in der angegebenen Größe erstellt.
Über die VSN kann dabei jede Platte eines importierten Pubsets ausgewählt werden.

### Zuweisen von Paging-Dateien zum Seitenwechselbereich

Eine existierende Paging-Datei kann über das Kommando EXTEND-PAGING-AREA in Betrieb genommen, d.h. dem Seitenwechselbereich zugeordnet werden. Damit sie auch in den folgenden Systemläufen von Beginn an verwendet wird, muss sie in die System-Parameterdatei eingetragen werden (siehe Parametersatz PAGING im Abschnitt „Auswahl des Seitenwechselbereichs beim Startup (PAGING)“ auf Seite 115).

Das Einrichten von Paging-Dateien kann auch mit der SIR-Anweisung CREATE-PAGING-FILE erfolgen; SIR ist im Handbuch „Dienstprogramme“ [15] beschrieben.

Es ist sinnvoll, bereits bei der Systeminstallation Paging-Dateien auf dem Home-Pubset einzurichten, da somit automatisch die Anforderungen des Betriebssystems sichergestellt sind.

Die benötigte Größe des Seitenwechselbereichs hängt von der Anzahl und der virtuellen Programmgröße der Programme ab, die gleichzeitig ablaufen. Hinzu kommt der jeweils generierte Systemadressraum.

Für eine erste Abschätzung wird folgende Formel empfohlen:

| benötigte Größe = 2 * (Anzahl Benutzertask * virtuelle Programmgröße + Systemadressraum) |
|---|

Nach Aufnahme des Betriebs sollte eine Überprüfung (z.B mit dem Kommando SHOW-PAGING-CONFIGURATION oder mit openSM2; siehe Handbuch „openSM2“ [51]) erfolgen, um die tatsächlich benötigte Größe des Seitenwechselbereichs festzustellen (siehe auch „Performance-Handbuch“ [39]).

#### Freigabe von Paging-Dateien aus dem Seitenwechselbereich

Mit dem Kommando REDUCE-PAGING-AREA kann im laufenden Betrieb eine Paging-Datei aus dem Seitenwechselbereich entnommen werden.

Die Freigabe ist aber nur dann möglich, wenn der vorhandene Seitenwechselbereich noch nicht ausgeschöpft ist und das System durch diese Reduzierung nicht in einen Sättigungszustand gerät.
Das Kommando REDUCE-PAGING-AREA läuft asynchron in einer Server-Task ab.

Zu Möglichkeiten zur beschleunigten Freigabe siehe Seite 180.

#### Löschen von Paging-Dateien

Zum Löschen einer Paging-Datei steht das Kommando DELETE-PAGING-FILE zur Verfügung. Der zugehörige Pubset muss dabei importiert sein.

Die Ausführung des Kommandos DELETE-PAGING-FILE dauert relativ lange, weil der Dateiinhalt gelöscht wird (Überschreiben mit binären Nullen).
Das Kommando läuft deshalb asynchron in einer Server-Task ab.

Eine Besonderheit besteht beim Löschen einer Paging-Datei des Home-Pubsets:
Das Löschen mit DELETE-PAGING-FILE wird abgelehnt, wenn nicht genügend andere Paging-Dateien im Home-Pubset erhalten bleiben.

#### Dynamische Rekonfiguration des Seitenwechselbereichs

Während des Systemlaufs kann der Seitenwechselbereich jederzeit mit dem Kommando EXTEND-PAGING-AREA dynamisch erweitert werden. Voraussetzung ist das Importieren des Pubsets mit der aufzunehmenden Paging-Datei und die Beachtung der max. Größe des Seitenwechselbereichs von 4 TB.

Mit dem Kommando REDUCE-PAGING-AREA kann der Seitenwechselbereich dynamisch verkleinert werden. Dabei sind einige Vorausetzungen zu beachten, z.B. ob die verbleibende Größe des Seitenwechselbereichs für die anstehenden Aufgaben ausreicht.



**ACHTUNG!**
Es wird empfohlen, mit der Rekonfiguration des Seitenwechselbereichs im laufenden Betrieb sehr sorgsam umzugehen und sie auf ein Mindestmaß zu beschränken. Häufiges Rekonfigurieren kann die Gesamt-Performance des Systems immer mehr verschlechtern. Dies gilt besonders bei Systemen mit hoher Paging-Last.

Folgende Arbeitsweise dient zur Erhaltung einer guten Gesamt-Performance:

- Beschränkung auf Paging-Dateien, die voraussichtlich nie oder selten aus dem Seitenwechselbereich genommen werden sollen.
- Die Verwendung zusammenhängender Paging-Dateien wird empfohlen.
- Es ist günstiger, viele kleine Platten zu verwenden als wenige große.
- Für diese Platten wird die exklusive Verwendung für Paging-Dateien empfohlen (reine Paging-Pubsets).

**Möglichkeiten zur Beschleunigung der Freigabe von Paging-Dateien**

Ist die freizugebende Paging-Datei fast vollständig ausgelastet, so ist durch die Verlagerung der virtuellen Seiten auf andere Paging-Dateien mit einer längeren Kommandoausführungsdauer, einer hohen I/O-Auslastung und mit einer hohen CPU-Belastung zu rechnen. Um diese Belastungen zu verringern bzw. die Reduktion zu erleichtern, sind folgende Möglichkeiten gegeben:

a) Die Reduktion wird dem Betriebssystem einige Zeit vorher angekündigt.

   Die betroffene Paging-Datei wird mit dem Kommando MODIFY-PAGING-AREA-ATTRIBUTES (Operand UTILIZATION=*LOW) ab diesem Zeitpunkt – so weit wie möglich – nicht mehr zur Aufnahme virtueller Seiten herangezogen.
   Durch Zugriff auf Seiten innerhalb der Paging-Datei oder Seitenfreigabe wird die Paging-Datei leerer und bei der eigentlichen Reduktion sind weniger Ein-/Ausgaben zum Auslagern dieser Seiten nötig.

   Ein Erfolg dieser Maßnahme ist aber nicht garantiert: Entweder kann auf das Einlagern von Seiten in diese Paging-Datei nicht verzichtet werden (wegen zu hoher Paging-Rate) oder auf die Seiten innerhalb der Paging-Datei wird nicht zugegriffen.

   Ob diese Maßnahme den gewünschten Erfolg bringt, kann über das Kommando SHOW-PAGING-CONFIGURATION ermittelt werden: Ein Ansteigen der ausgegebenen freien Größe der betroffenen Paging-Datei ist ein Maß für den Erfolg und sagt aus, wann die Reduzierung zu starten ist. Dies kann nach einigen Minuten oder auch erst Stunden der Fall sein.

b) Dem Betriebssystem wird bereits beim Zuweisen der Paging-Datei zum Seitenwechselbereich mit dem Kommando EXTEND-PAGING-AREA ...,UTILIZATION=*LOW mitgeteilt, dass diese Paging-Datei zur Aufnahme der virtuellen Seiten weniger herangezogen werden soll.
   Es gelten die gleichen Aussagen zum Erfolg wie im vorhergehenden Punkt.

c) Beim Kommando EXTEND-PAGING-AREA gibt es bei Angabe des Operanden LATER-REDUCTION=*REQUESTED die Möglichkeit, die CPU-Belastung bei der Reduktion zu verringern.
Für die Verwaltung dieser Paging-Dateien wird jedoch mehr residenter Arbeitsspeicher verbraucht.

Bei allen genannten Maßnahmen gilt die Empfehlung, sie auf maximal zwei Paging-Dateien anzuwenden.

### 4.4.2 Auswahl des Seitenwechselbereichs bei Startup

Die Systembetreuung kann zum Startup-Zeitpunkt über den Parameterservice die Datenträger bestimmen, die für den Seitenwechsel benutzt werden sollen. Dabei ist zu beachten, dass auf diesen Datenträgern Paging-Dateien `SYS.PAGING.<vsn>` angelegt sind und dass die maximale Anzahl von 128 Datenträgern nicht überschritten wird.

Der von Startup initialisierte Seitenwechselbereich sollte eine Mindestgröße von 200 MB haben. Ist der Seitenwechselbereich kleiner als 200 MB, wird als Warnung die Meldung `NSI5115` an der Konsole ausgegeben; die Systemeinleitung wird fortgesetzt.

Hat die Systembetreuung keine Parameter für die Auswahl von Paging-Dateien angegeben, dann werden automatisch alle auf dem Home-Pubset eingerichteten Paging-Dateien `SYS.PAGING.<vsn>` für den Seitenwechsel verwendet.
Mit dem Parameterservice können Datenträger für Seitenwechsel innerhalb und außerhalb des Home-Pubsets angegeben werden.
Die dazu notwendigen Anweisungen muss die Systembetreuung in der Parameterdatei hinterlegen (siehe PAGING-Parametersatz, Kapitel „Parameterservice“ auf Seite 75).

## 4.5 Maßnahmen zur Verhinderung von Sättigungszuständen

In folgenden Bereichen kann die Ausführung einer Speicherplatzanforderung zu Sättigungszuständen führen: Hauptspeicher, Systemadressraum und Seitenwechselspeicher.

Speicherengpässe können auch bei folgenden Gelegenheiten auftreten, die an anderer Stelle beschrieben werden:

- bei zu hoher Hauptspeicher-Auslastung oder zu hoher Verdrängungsrate (siehe Seite 483)
- bei der Anforderung von Speicherplatz auf gemeinschaftlichen Datenträgern (siehe Seite 393)

Die Meldungen, die abhängig von Sättigungsursache und -zustand ausgegeben werden, sind in den folgenden Abschnitten aufgeführt. Das Gleiche gilt auch für die Maßnahmen, die sowohl vom Operator als auch von der Systemverwaltung durchgeführt werden können. In diesem Abschnitt sollen die langfristigen Maßnahmen zur Verhinderung von Sättigungszuständen im Vordergrund stehen.

#### Hauptspeicher

- Die Hauptspeichergröße sollte entsprechend den Empfehlungen des Performance-Handbuchs gewählt werden.
  Ein zu kleiner Hauptspeicher wirkt sich meist sehr negativ auf die Performance aus, weil die Paging-Rate und die Verdrängungsrate zu hoch werden.
- Bei VM2000-Gastsystemen müssen die Empfehlungen des Performance-Handbuchs für den Wert von MINIMAL-MEMORY-SIZE eingehalten werden.
- Mit dem Kommando MODIFY-SYSTEM-BIAS kann Benutzerprogrammen erlaubt werden, eine größere Anzahl von Benutzerseiten resident zu machen.
  Es wird empfohlen, diese Erlaubnis auf die Benutzerkennungen zu begrenzen, in denen solche Programme laufen müssen (Kommandos ADD-USER und MODIFY-USER-ATTRIBUTES).

### Systemadressraum

- Der Bedarf an Systemadressraum richtet sich stark nach der System-Konfiguration und der Geräte-Peripherie, nach den eingesetzten Subsystemen und nach der Last.
- Es wird dringend empfohlen, den Systemadressraum großzügig zu bemessen (die Größe ist nur bei S-Servern einstellbar, siehe Seite 185).
- Ein zu geringer Vorhalt kann unerwünschte Betriebsunterbrechungen erzwingen (z.B. wegen Subsystem-Versionsumstieg, Anstieg der Last, Ausbau der Systemkonfiguration oder wegen der Nachwirkungen eines Sättigungszustands).
- Empfohlen wird auch die Nutzung der vorhandenen Möglichkeiten zur Lastbegrenzung. So kann z.B. die Anzahl der Benutzer in den Jobklassen begrenzt werden (Kommando MODIFY-JOB-CLASS). Auch die Programmgröße kann benutzerspezifisch beschränkt werden (über die Kommandos ADD-USER und MODIFY-USER-ATTRIBUTES).

### Seitenwechselspeicher

- Die Größe des Seitenwechselspeichers sollte entsprechend den Empfehlungen des Performance-Handbuchs eingestellt werden.
- Ein ausreichend großer Seitenwechselspeicher muss bereits bei der Systemeinleitung in Betrieb genommen werden (Parameterservice).
- Falls der Bedarf an Seitenwechselspeicher während des Systemlaufs stark anwächst, sollte die Erweiterung möglichst frühzeitig erfolgen (über das Kommando EXTEND-PAGING-AREA).
- Die Möglichkeiten zur Lastbegrenzung sollten genutzt werden (siehe oben bei „Systemadressraum“)

Um einen Leistungsabfall zu verhindern, treten bei Sättigungszuständen Algorithmen in Kraft, die der Sättigung entgegenwirken.

Vom Organisationsprogramm werden dabei die in den folgenden Tabellen enthaltenen Meldungen auf der Bedienstation ausgegeben.

### 4.5.1 Hauptspeicher

#### Sättigungszustand

Meldung: `EXC0870 CORE SATURATION LEVEL=i`

`i` ist eine Ziffer von 0 bis 3 und bezeichnet den Grad der Sättigung.

| Grad | Auswirkungen im System | Maßnahme des Operators |
|---|---|---|
| 0 | Normaler Systemablauf | Keine |
| 1 | Neue Batch-Jobs werden nicht gestartet. | Keine |
| 2 | Neue Batch-Jobs werden nicht gestartet. | Keine |
| 3 | Neue Batch-Jobs werden nicht gestartet.<br>Die Anforderung von residenten Speicherseiten wird nur noch bei besonders wichtigen System-Jobs befriedigt. | Der Operator kann Jobs abbrechen. |

#### Rückgang der Sättigung

| Grad | Auswirkungen im System | Maßnahme des Operators |
|---|---|---|
| 2 ... 0 | Rückgang der Sättigung<br>Nach Erreichen von Level=0 können wieder Batch-Jobs gestartet werden. | Keine |

#### Maßnahmen zur Verhinderung der Sättigung

Das Auftreten eines Sättigungszustandes zeigt an, dass das System überlastet ist. Wenn der Hauptspeicherbedarf sich nicht auf Dauer verringern lässt, benötigt das System einen größeren Hauptspeicher. Handelt es sich um ein VM2000-Gastsystem, so ist eine Erhöhung der MINIMAL-MEMORY-SIZE durch den VM2000-Administrator erforderlich. Eine größere NEW-MEMORY-SIZE bei gleichbleibender MINIMAL-MEMORY-SIZE ist als Gegenmaßnahme ungeeignet.

Bei SQ-Servern ist hier der BS2000-Speicher gemeint. Welcher Anteil des gesamten Speichers das ist, wird über X2000 eingestellt.

*Beispiele zur Verringerung des Hauptspeicherbedarfs eines Systems*

- DAB-Cache-Bereiche im Hauptspeicher verkleinern
- Anzahl der residenten Katalog-Puffer einzelner Pubsets gezielt verringern oder die Puffer seitenwechselbar machen
- Nicht belegte Pubsets exportieren
- Entladen von Anwendungen, die Benutzerseiten resident gesetzt haben
- Entladen großer Anwendungen bzw. Programme
- Deutliche Verringerung der Taskanzahl im System

## 4.5.2 Systemadressraum

### Sättigungszustand

Meldung: `EXC0874 ADDRESS SPACE SATURATION=i`

`i` ist eine Ziffer von 0 bis 2 und bezeichnet den Grad der Sättigung.

<table>
<tr><th>Grad</th><th>Auswirkungen im System</th><th>Maßnahme des Operators</th></tr>
<tr><td>0</td><td>Normaler Systemablauf</td><td>Keine</td></tr>
<tr><td>1</td><td>Es werden keine neuen Batch-Jobs gestartet.</td><td rowspan="2">Systemverwaltung verständigen</td></tr>
<tr><td>2</td><td>Es werden keine neuen Batch-Jobs gestartet.</td></tr>
</table>

### Maßnahmen zur Verhinderung der Sättigung

Das Auftreten eines Sättigungszustandes zeigt an, dass das System überlastet ist. Wenn das System immer wieder in einen Sättigungszustand kommt, ist es nicht ausreichend konfiguriert.
Dadurch kann es notwendig werden, den Benutzeradressraum zu verkleinern (siehe auch Handbuch „Systeminstallation“ [57], Anpassen des BS2000-Organisationsprogramms, Prozedur SYSPRC.BS2000-EXEC.180).

An SQ-Servern bestehen diese Konfigurationsmöglichkeiten nicht.

*Beispiele zur Verringerung des Systemadressraumbedarfs eines Systems*

- Anzahl der Katalog-Puffer einzelner Pubsets gezielt verringern.
- Nicht belegte Pubsets exportieren.
- Entladen großer Anwendungen bzw. Programme.
- Deutliche Verringerung der Taskanzahl im System.
- Vor dem Entladen großer Subsysteme wird ausdrücklich gewarnt. Hierdurch kann zwar der Sättigungszustand meist kurzfristig aufgehoben werden, häufig lässt sich aber das entladene Subsystem auch später nicht mehr laden, sodass für den Rest des Systemlaufs darauf verzichtet werden muss.
- Häufige Sättigung im Systemadressraum trotz richtiger Konfigurierung kann auf einen Systemfehler hinweisen. Bitte konsultieren Sie den Service (Systemdiagnose).

### 4.5.3 Seitenwechselspeicher

**Sättigungszustand**

Meldung: `EXC0873 NEW PAGING MEMORY SATURATION LEVEL=i`

`i` ist eine Ziffer von 0 bis 3 und bezeichnet den Grad der Sättigung.

| Grad | Auswirkungen im System | Maßnahme des Operators |
|---|---|---|
| 0 | Normaler Systemablauf | Keine |
| 2 | Batch-Jobs, die Klasse-6-Speicher anfordern, werden ausgelagert. [1)]<br><br>Neue Jobs werden nicht gestartet. | Der Operator muss versuchen, Jobs abzubrechen oder mit EXTEND-PAGING-AREA den Seitenwechselbereich dynamisch zu erweitern. |
| 3 | Dialog- und Batch-Jobs, die Klasse-6-Speicher anfordern, werden ausgelagert.<br><br>Folgende Meldung erscheint auch auf SYSOUT der Dialog-Jobs.<br>`EXC0844 TASK TEMPORARILY HELD DUE TO PAGING DEVICE SATURATION. TASK WILL BE CONTINUED AUTOMATICALLY`<br><br>Die TCB-Adressen der Jobs werden in einer Warteschlange verwaltet. Sie werden dort wie folgt eingereiht:<br>– Dialog-Jobs vor Batch-Jobs<br>– Jobs mit hoher Priorität vor Jobs mit niederer Priorität<br>– Jobs gleicher Priorität abhängig von der Zahl ausgelagerter Seiten (Jobs mit weniger ausgelagerten Seiten kommen zuerst)<br><br>Neue Jobs werden nicht gestartet. | Der Operator muss versuchen, Jobs abzubrechen oder mit EXTEND-PAGING-AREA den Seitenwechselbereich dynamisch zu erweitern.<br>Es kann zum Systemstillstand bzw. Systemabbruch kommen. |

1) Der Klasse-6-Speicher des Jobs wird in eine temporäre Plattendatei (EAM) gerettet und dann freigegeben. Ein solcher Job muss folgende Bedingungen erfüllen:

- Die Klasse-6-Speicheranforderung bezieht sich nicht auf gemeinsamen Speicherplatz für mehrere Jobs (Memory Pool).
- Der Job ist an keine Serialisierungskennung angeschlossen.
- Der Job hat keine Datei-Seiten gesperrt.

### Rückgang der Sättigung

| Grad | Auswirkungen im System | Maßnahme des Operators |
|---|---|---|
| 1 | Level=1 tritt nur bei Rückgang der Sättigung auf und zeigt lediglich an, dass der erste Job in der Warteschlange wieder eingelagert wurde, es aber noch weitere ausgelagerte Jobs gibt. | Keine |
| 0 | Alle ausgelagerten Jobs werden wieder bedient.<br>Es können wieder Jobs gestartet werden. | Keine |

### Maßnahmen zur Verhinderung der Sättigung

Das Auftreten eines Sättigungszustandes zeigt an, dass das System überlastet ist.

Wenn das System immer wieder in einen Sättigungszustand kommt, ist der Seitenwechselspeicher zu klein oder die Maßnahmen zur Lastbegrenzung waren unzureichend.

- Vergrößern Sie den Seitenwechselspeicher.
- Sorgen Sie dafür, dass nachfolgende Systemläufe bei der Systemeinleitung einen ausreichend großen Seitenwechselspeicher in Betrieb nehmen (Parameterservice).
- Überprüfen Sie Ihre Maßnahmen zur Lastbegrenzung.

*Beispiele zur Verringerung des Seitenwechselspeicherbedarfs eines Systems (falls Erweiterung nicht gewünscht wird)*

- Entladen großer Anwendungen bzw. Programme
- Deutliche Verringerung der Taskanzahl im System

# 5 Geräteverwaltung

Die Geräteverwaltung von BS2000/OSD (Nucleus Device Management, NDM) verwaltet die periphere Konfiguration des Servers. Die Kanalperipherie der S-Server wird bei der Hardware-Generierung mit IOGEN (siehe Handbuch „Systeminstallation“ [57]) vorgegeben. Die Busperipherie der SQ-Server wird über X2000 bekannt gemacht.

Die Geräteverwaltung umfasst folgende Funktionen:

- Sie steuert die Verfügbarkeit aller Hardware-Einheiten und deren Verbindungen für das System.
- Sie stellt Belegungs- und Freigabemechanismen für Geräte und Datenträger zur Verfügung.
- Sie wickelt Montieraufträge für Datenträger ab und schützt diese Datenträger vor Fehleingriffen.
- Sie ermöglicht die Reservierung von Geräten, Datenträgern und Dateien unter Berücksichtigung von Auftragsprioritäten.
- Sie informiert über Belegungs- und Verfügbarkeitszustände der Konfiguration.

In diesem Kapitel werden die Server-Konfiguration und -Rekonfiguration und die dynamische I/O-Konfigurationsänderung für S-Server ausführlich behandelt. Für SQ-Server werden jeweils die Besonderheiten herausgestellt.

Danach werden die Komponente NDM (Nucleus Device Management) zur Betriebsmittelbelegung und -reservierung sowie die Überwachung, Auswahl und Bedienung von gemeinschaftlichen Datenträgern beschrieben.

Es folgen detaillierte Informationen zur Fehlerbehandlung ausgewählter Gerätesteuerungen und zur Verwaltung privater Datenträger. Danach folgen Hinweise zu den Dienstprogrammen IORM und SANCHECK.

## 5.1 Bestandteile von Konfigurationen

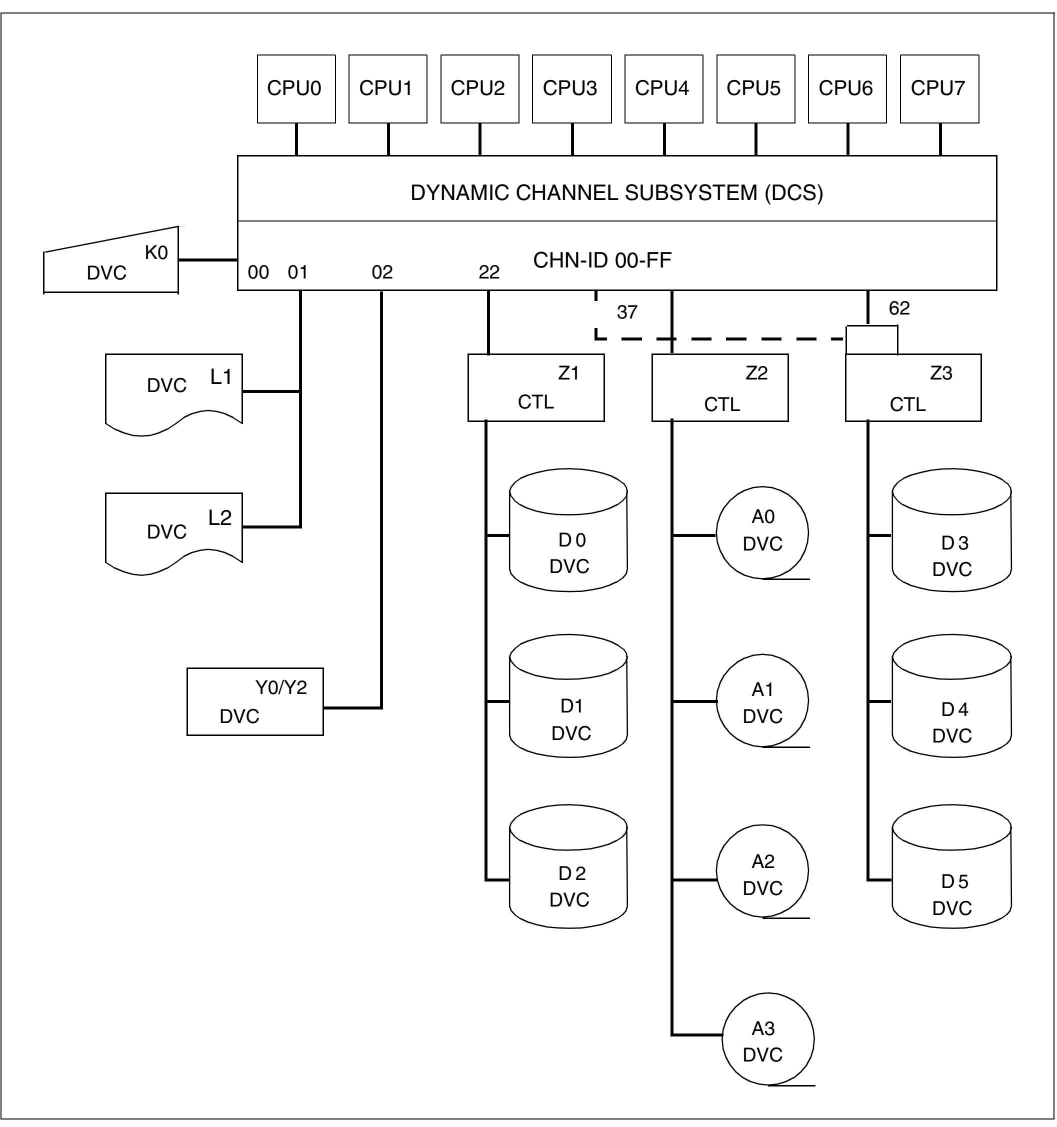



Bild 9: Ausschnitt einer Server-Konfiguration (S-Server)

| **Kommando** | **Bedeutung** |
|---|---|
| ADD-IO-UNIT | neue Ein-/Ausgabe-Einheiten dynamisch hinzufügen [1] |
| ATTACH-DEVICE | Zuschalten einer Hardware-Einheit,<br>bei Kanälen und Controllern lokal oder VM2000-global |
| DETACH-DEVICE | Wegschalten einer Hardware-Einheit,<br>bei Kanälen und Controllern lokal oder VM2000-global |
| INCLUDE-DEVICE-CONNECTION | Zuschalten einer logischen Verbindung lokal oder VM2000-global |
| MODIFY-IO-UNIT | Pfadbeschreibung einer Steuerung ändern, [1]<br>bevorzugten Pfad eines Gerätes ändern, [1]<br>bevorzugtes PAV-Gerät definieren,<br>Timeout-Einstellungen ändern [2] |
| REMOVE-DEVICE-CONNECTION | Wegschalten einer logischen Verbindung lokal oder VM2000-global |
| REMOVE-IO-UNIT | Ein-/Ausgabe-Einheiten dynamisch entfernen [1] |
| SECURE-RESOURCE-ALLOCATION | Anfordern von Betriebsmitteln |
| SHOW-DEVICE-CONFIGURATION | Abfragen der Ein-/Ausgabe-Konfiguration |
| START-CONFIGURATION-UPDATE | dynamische I/O-Konfigurationsänderung einleiten |
| STOP-CONFIGURATION-UPDATE | dynamische I/O-Konfigurationsänderung beenden,<br>aktuelles IORSF sichern |
| UNLOCK-DEVICE | Rücksetzen der Hardware-Reservierung |
| **Makro** | **Bedeutung** |
| NKDINF | Informieren über den Belegungs- und Verfügbarkeitszustand der (peripheren) Konfiguration |

Tabelle 12: Schnittstellenübersicht zur Konfiguration

[1] Diese Funktionen sind nur verfügbar, wenn eine dynamische I/O-Konfigurationsänderung an S-Servern eingeleitet ist.

[2] Diese Funktion ist auch außerhalb einer dynamischen Rekonfiguration unabhängig vom Server-Typ verfügbar.

### 5.1.1 Hardware-Einheiten

**Arten von Hardware-Einheiten**

In einer Konfiguration gibt es folgende Arten von Hardware-Einheiten:

- die CPUs (Central Processor Unit)
- die Kanäle (Channel, CHN), die mit der IOGEN-Anweisung CHN definiert werden
- die Mehrgerätesteuerungen (Controller, CTL), die mit der IOGEN-Anweisung CTL festgelegt werden (Band- und Plattensteuerungen)
- die Geräte (Device, DVC), definiert mit der IOGEN-Anweisung DVC

i Auf SQ-Servern sind für Bus- und FC-Peripherie in BS2000/OSD nur Kanäle (CHN) und Geräte (DVC), jedoch keine Controller (CTL) bekannt und sichtbar.

Man unterscheidet bezüglich der logischen Verbindungen zwischen inneren und äußeren Einheiten, die der Ein-/Ausgabe dienen.

Die Kanäle (CHN) sind die innersten, die Geräte (DVC) die äußersten Hardware-Einheiten. Die verschiedenen Hardware-Einheiten stehen, von innen nach außen, in der Reihenfolge CHN – CTL – DVC

Von zwei logisch direkt verbundenen Hardware-Einheiten, z.B. CTL – DVC, ist CTL die innere, DVC die äußere Einheit.

Bezogen auf Bild 9 auf Seite 190 bedeutet das:

| | | |
|---|---|---|
| CHN 00 | ist die innere Einheit zu | DVC K0 |
| CTL Z1 | ist die äußere Einheit zu | CHN 22 |
| CTL Z2 | ist die innere Einheit zu | DVC A0,A1,A2,A3 |
| DVC D0,D1,D2 | sind die äußeren Einheiten zu | CTL Z1 |
| CTL Z3 | ist die äußere Einheit zu | CHN 62,37 |

Voraussetzungen:

- CHN 00 hat die channel-path-id 00
- CHN 22 hat die channel-path-id 22
- CTL Z1 ist angeschlossen an channel-path-id 22
- CTL Z3 ist angeschlossen an channel-path-id 37 und 62

### Zustände von Hardware-Einheiten

Diese Hardware-Einheiten können folgende Zustände haben:

ATTACHED

Die Hardware-Einheit ist zum System zugeschaltet und kann für Ein-/Ausgaben benutzt werden. Dieser Zustand ist eingestellt:

- nach der Systemeinleitung, wenn die entsprechende Einheit bei der Hardware-Generierung mit IOGEN mit dem Attribut „A“ generiert wurde (z.B. DVC A0,CB,**A**,(...))
- nach der Systemeinleitung, wenn die entsprechende Einheit im Startup-Parameterservice mit der IOCONF-Anweisung MOD-IO-UNIT ..., STATE=ATT geändert wurde
- nach erfolgreichem Aufruf des Kommandos ATTACH-DEVICE (ATT)
- nach erfolgreichem Aufruf des Kommandos ADD-IO-UNIT (mit STATE=*ATTACHED)

ATTACH-PENDING

Für Platten- oder Bandgeräte erfolgt eine ATTACH-Bearbeitung. Die Hardware-Einheit kann noch nicht für Ein-/Ausgaben verwendet werden. Wird der Rekonfigurationsauftrag ausgeführt, geht die Hardware-Einheit in den Zustand ATTACHED über.
Den Zustand DETACHED erhält die Hardware-Einheit, wenn der Rekonfigurationsauftrag nicht ausgeführt werden kann (vergleiche Abschnitt „Besonderheiten für MBK-Geräte“ auf Seite 204).

Dieser Zustand ist während der Durchführung von Ein-/Ausgaben zur Ermittlung von Steuerungs- und Gerätedaten sowie zur Reservierung von MBK-Geräten eingestellt.

DETACHED

Die Hardware-Einheit ist vom System weggeschaltet und kann nicht für Ein-/Ausgaben benutzt werden. Der Zustand DETACHED wird weiter unterteilt in:

- DETACHED EXPLICITLY. Dieser Zustand ist eingestellt:
  - nach der Systemeinleitung, wenn die entsprechende Einheit bei der Hardware-Generierung mit IOGEN mit dem Attribut „D“ generiert wurde (z.B. DVC_A0,CB,**D**,(...))
  - nach der Systemeinleitung, wenn die entsprechende Einheit im Startup-Parameterservice mit der IOCONF-Anweisung MOD-IO-UNIT ..., STATE=DET geändert wurde
  - nach einem erfolgreichen Aufruf des Kommandos DETACH-DEVICE (DET)
  - nach einer automatischen Rekonfiguration durch das Betriebssystem
  - nach der Systemeinleitung, wenn eine Hardware-Einheit als „offline“ erkannt wird

- DETACHED IMPLICITLY
  Alle inneren Verbindungen der entsprechenden Einheit befinden sich im Zustand REMOVED.

DETACH-PENDING

Die Einheit ist noch von einer Task oder vom System belegt und steht nach Belegungsende nicht mehr zur Verfügung. Weitere Belegungsversuche werden nicht mehr zugelassen. Der Zustand DETACH-PENDING tritt i.A. nur während einer angegebenen Wartezeit (Operand FORCE=*NO) im DETACH-DEVICE-Kommando auf. Er kann auch dann auftreten, wenn ein DETACH-DEVICE-Kommando mit Operand FORCE=*YES für ein Magnetbandkassetten-Gerät abgesetzt wurde (vergleiche Abschnitt „Besonderheiten für MBK-Geräte“ auf Seite 204).
Kann der Rekonfigurationsauftrag während der Wartezeit durchgeführt werden, geht die Einheit in den Zustand DETACHED über, sonst wird (wieder) der Zustand ATTACHED eingestellt. Der Zustand DETACH-PENDING wird unterteilt in:

- DETACH-PENDING EXPLICITLY
  Dieser Zustand wird eingestellt, wenn die Einheit direkt weggeschaltet wird.
- DETACH-PENDING IMPLICITLY
  Dieser Zustand wird eingestellt, wenn keine innere Verbindungen der Einheit im Zustand INCLUDED und zumindest eine im Zustand REMOVE-PENDING ist.

INVALID

Das Gerät kann nicht benutzt und auch nicht rekonfiguriert (zugeschaltet) werden.
Beispiel: Das Gerät wurde mit REMOVE-IO-UNIT entfernt.

### 5.1.2 Logische, rekonfigurierbare Verbindungen

**Arten von logischen, rekonfigurierbaren Verbindungen**

In einer Konfiguration bestehen folgende logische (rekonfigurierbare) Verbindungen:

- zwischen den Kanälen und den direkt daran angeschlossenen Geräten (DVC, z.B. Bedienstationen, Drucker, Magnetbandgeräte)
- zwischen den Kanälen und den daran angeschlossenen Mehrgerätesteuerungen (CTL, Platten- und Magnetbandsteuerungen)
- zwischen den Steuerungen und den daran angeschlossenen Geräten (DVC, Magnetbandgeräte und Platten)

Logische Verbindungen bestehen zum Beispiel (siehe Bild 9 auf Seite 190):

| | | | |
|---|---|---|---|
| zwischen dem Kanal | 00 | und dem Gerät | K0 |
| zwischen dem Kanal | 22 | und der Steuerung | Z1 |
| zwischen der Steuerung | Z2 | und den Geräten | A0,A1,A2,A3 |
| zwischen der Steuerung | Z3 | und den Kanälen | 37 und 62 |

Keine logische Verbindung besteht jedoch zwischen Hardware-Einheiten, die nicht direkt miteinander verbunden sind, z.B. zwischen dem Kanal 22 und dem Plattenspeicher D0, da zwischen beiden noch eine Plattensteuerung ist.

Man unterscheidet innere und äußere logische Verbindungen:

- Eine innere Verbindung geht immer in Richtung zu einer inneren Einheit (z.B. von DVC zu CTL oder von CTL zu CHN).
- Eine äußere Verbindung geht in Richtung zu einer äußeren Einheit (CHN zu CTL oder CTL zu DVC).

Die innere Verbindung einer Einheit ist gleichzeitig die äußere Verbindung einer anderen Einheit. Die logische Verbindung zwischen der Steuerung Z1 und dem Plattenspeicher D0 ist von Z1 aus gesehen eine äußere, von D0 aus dagegen eine innere Verbindung.

### Zustände von logischen, rekonfigurierbaren Verbindungen

INCLUDED

Die logische Verbindung ist zum System zugeschaltet und kann für Ein-/Ausgaben benutzt werden. Dieser Zustand ist eingestellt:

- nach der Systemeinleitung, wenn die entsprechende Einheit im Startup-Parameterservice mit der IOCONF-Anweisung MOD-IO-UNIT *CON(..., STATE=INC) geändert wurde
- nach erfolgreichem Aufruf des Kommandos INCLUDE-DEVICE-CONNECTION (INC)

REMOVED

Die logische Verbindung ist vom System weggeschaltet und kann nicht für Ein-/Ausgaben benutzt werden. Der Zustand REMOVED wird weiter unterteilt in:

- REMOVED EXPLICITLY
  Dieser Zustand wird eingestellt:
  - nach der Systemeinleitung, wenn die entsprechende Einheit im Startup-Parameterservice mit der IOCONF-Anweisung MOD-IO-UNIT *CON(..., STATE=REM) geändert wurde
  - nach einer automatischen Rekonfiguration durch das Betriebssystem
  - nach erfolgreichem Aufruf des Kommandos REMOVE-DEVICE-CONNECTION
- REMOVED IMPLICITLY
  Dieser Zustand wird eingestellt, wenn die innere Hardware-Einheit im Zustand DETACHED ist.

REMOVE-PENDING

Die logische Verbindung ist noch belegt und steht danach dem System nicht mehr zur Verfügung. Dieser Zustand tritt nur während einer Wartezeit in den Kommandos REMOVE-DEVICE-CONNECTION bzw. DETACH-DEVICE auf. Kann der Rekonfigurationsauftrag durchgeführt werden, geht die Verbindung in den Zustand REMOVED über, sonst wird (wieder) der Zustand INCLUDED eingestellt. REMOVE-PENDING wird unterteilt in:

- REMOVE-PENDING EXPLICITLY
  Dieser Zustand wird eingestellt, wenn die Verbindung direkt (mit Wartezeit: Operand FORCE=NO im Kommando REMOVE-DEVICE-CONNECTION) weggeschaltet wird.
- REMOVE-PENDING IMPLICITLY
  Dieser Zustand wird eingestellt, wenn die innere Einheit dieser Verbindung im Zustand DETACH-PENDING ist.

## 5.2 Rekonfiguration

Unter Rekonfiguration versteht man das Zu- oder Wegschalten von Komponenten der Server-Konfiguration. Eine Konfiguration besteht aus Hardware-Einheiten und deren logischen Verbindungen (siehe Bild 9 auf Seite 190).

Dazu stehen dem Operator folgende sog. Rekonfigurationskommandos zur Verfügung:

| Kommando | Bedeutung |
|---|---|
| ATTACH-DEVICE | Zuschalten einer Hardware-Einheit,<br>bei Kanälen und Controllern lokal oder VM2000-global |
| DETACH-DEVICE | Wegschalten einer Hardware-Einheit,<br>bei Kanälen und Controllern lokal oder VM2000-global |
| INCLUDE-DEVICE-CONNECTION | Zuschalten einer logischen Verbindung lokal oder VM2000-global |
| REMOVE-DEVICE-CONNECTION | Wegschalten einer logischen Verbindung lokal oder VM2000-global |

Tabelle 13: Rekonfigurationskommandos

Weitere für die Rekonfiguration relevante Kommandos und Makros sind der Tabelle 15 auf Seite 216 zu entnehmen.

### 5.2.1 Rekonfiguration bei Multiprozessoren

Das System reagiert automatisch auf Hardware-Ausfälle. Wenn eine Komponente (CPU, Channel) ausfällt, wird sie vom Betriebssystem weggeschaltet. Der Betrieb wird auf den verbleibenden Komponenten weitergeführt.

Der Operator kann Komponenten mittels Rekonfigurationskommandos zu- bzw. wegschalten. Wenn eine Komponente zu Wartungszwecken oder aus anderen Gründen deaktiviert werden soll, kann sie der Operator mit dem DETACH-DEVICE-Kommando wegschalten. Der Restbetrieb wird dann auf den verbleibenden Komponenten weitergeführt. Soll eine weggeschaltete Komponente wieder verfügbar gemacht werden, kann sie der Operator mit dem Kommando ATTACH-DEVICE zuschalten.

Die Rekonfiguration schlägt fehl, wenn einer der folgenden Umstände vorliegt:

- nach dem Wegschalten einer CPU ist nicht noch mindestens eine CPU arbeitsfähig
- bei Ausfall einer CPU sind Daten des Betriebssystems inkonsistent und die Hardware konnte keinen gültigen Hardware-Kontext mehr zur Verfügung stellen
- gemeinschaftliche Datenträger oder die letzte Bedienstation sind durch den Ausfall betroffen

Bei bestimmten Geräten muss nach einer Rekonfiguration ein Umschalten bzw. Einschalten durchgeführt werden, damit sie wieder vom System angesprochen werden können. Mit Umschalten ist in diesem Fall nicht nur das physikalische Umschalten des Gerätes auf einen Alternativpfad gemeint, obwohl dies eine Voraussetzung dafür ist. Umschalten beinhaltet alle Aktionen die erforderlich sind, damit das System die Geräte wieder ansprechen kann. Um- bzw. Einschalten liegt in der Verantwortung des Operators.



Tritt bei Servern mit mehr als einer CPU in einer der zugeschalteten CPUs ein Fehler auf, so wird in vielen Fällen der Server automatisch rekonfiguriert. Nur in den Fällen, bei denen ein gültiger Programm-Kontext der ausgefallenen CPU nicht hergestellt werden kann und der Lock-Mechanismus des Betriebssystems aktiv ist, kann keine Rekonfiguration durchgeführt werden.

## 5.2.2 Weg- und Zuschalten der Komponenten CPU, CHN, CTL und DVC

### Wegschalten von Komponenten durch das Betriebssystem

Bei Ausfall werden die Komponenten automatisch weggeschaltet. Der Operator erhält dann folgende Meldung an der Hauptbedienstation:

```
NKR0048    'CHANNEL=mn' DETACHED BY SYSTEM **********
```
(für CHN)

```
ETMRK48    CPUxy DETACHED BY SYSTEM **********
```
(für CPU)

mn Mnemotechnischer Gerätename der Komponente

Die Meldung `ETMRK48` muss vom Operator quittiert werden. Dies soll verhindern, dass ein Wegschalten übersehen wird.

Die Meldung wird an der Hauptbedienstation ausgegeben, die am intakten Ein-/Ausgabe-Prozessor angeschlossen ist.

Das (Wieder-)Zuschalten erfolgt mittels ATTACH-DEVICE durch den Operator, ist aber nicht in jedem Fall möglich.

### Wegschalten von Komponenten durch den Operator

Der Operator kann Komponenten mit dem Kommando DETACH-DEVICE (siehe Handbuch „Kommandos“ [27]) wegschalten.

Das ordnungsgemäße Wegschalten bestätigt das Betriebssystem für jede Komponente mit folgender Meldung an der Bedienstation:

```
NKR0045    'CHANNEL=mn' DETACHED **********
```
(für CHN)

```
ETMRK19    CPUxy DETACHED **********
```
(für CPU)

Das Kommando DETACH-DEVICE wird mit einer entsprechenden Meldung zurückgewiesen, wenn es momentan nicht ausgeführt werden kann.
Bei Kanälen und Gerätesteuerungen unter VM2000 auf S-Servern kann im Monitorsystem das Wegschalten nur im lokalen System oder in allen BS2000-Gastsystemen (VM2000-global) erfolgen. Das Kommando muss dann im Monitorsystem eingegeben werden.

Das (Wieder-)Zuschalten erfolgt mittels ATTACH-DEVICE.

### Zuschalten von Komponenten durch den Operator

Der Operator kann Komponenten in folgenden Fällen mittels ATTACH-DEVICE zuschalten:

- wenn eine Komponente nach Ausfall oder Wartung wieder verfügbar ist
- wenn die Systemeinleitung ohne eine Komponente durchgeführt wurde

Bei Kanälen und Gerätesteuerungen unter VM2000 kann im Monitorsystem das Zuschalten nur im lokalen System oder in allen BS2000-Gastsystemen (VM2000-global) erfolgen.

Das Zuschalten bestätigt das Betriebssystem für jede Komponente durch folgende Meldung an der Bedienstation:

| | | |
|---|---|---|
| `NKR0040` | `'<unit-class>=mn' ATTACHED` | (für CHN, CTL, DVC) |
| `ETMRK18` | `CPUxy ATTACHED` | (für CPU) |

### Zu- und Wegschalten von Extra-CPUs

Extra-CPUs gibt es an S- und SQ-Servern.

Der Operator kann eine oder mehrere Extra-CPUs im laufenden Betrieb zuschalten (Kommando ATTACH-DEVICE UNIT=*EXTRA-CPU(CPU-ID=...)).

Das Wegschalten der Extra-CPUs erfolgt mit dem Kommando DETACH-DEVICE.

Für ausführliche Informationen zum Zu- und Wegschalten von Extra-CPUs siehe Seite 214.

### 5.2.3 Wirkung der Rekonfigurationskommandos

Ein Rekonfigurationskommando bleibt ohne Wirkung, wenn ein Zustand eingestellt werden soll, der bereits eingestellt ist oder der nicht eingestellt werden kann. Es werden entsprechende Meldungen ausgegeben.

Das Zu- und Wegschalten von Kanälen, Steuerungen und Verbindungen unter VM2000 kann im Monitorsystem nur für das lokale System oder für alle BS2000-Gastsysteme (VM2000-global) erfolgen.
Dafür wurde der Operand SCOPE=OWN-SYSTEM-ONLY/VM2000-GLOBAL eingeführt.

**ATTACH-DEVICE**

Wird ein gültiges Kommando ATTACH-DEVICE eingegeben, so hat das folgende Auswirkungen:

Waren die angegebenen Hardware-Einheiten im Zustand DETACHED EXPLICITLY, so werden sie ATTACHED. Die Hardware-Einheiten können wieder benutzt werden.

Waren die zu diesen Hardware-Einheiten gehörenden äußeren Verbindungen REMOVED IMPLICITLY, so werden sie INCLUDED. Die logischen Verbindungen können wieder benutzt werden.

Die mit diesen Einheiten verbundenen äußeren Hardware-Einheiten, die sich im Zustand DETACHED IMPLICITLY befanden, werden ATTACHED, vorausgesetzt, mindestens eine logische Verbindung zu der äußeren Einheit ist im Zustand INCLUDED.

**DETACH-DEVICE**

Ein gültiges Kommando DETACH-DEVICE wirkt sich folgendermaßen aus:

Jede der angegebenen Hardware-Einheiten geht in den Zustand DETACHED EXPLICITLY über. Sie können nicht benutzt werden.

Alle zu diesen Hardware-Einheiten gehörenden äußeren Verbindungen erhalten den Zustand REMOVED IMPLICITLY. Sie können nicht benützt werden.

Alle mit diesen Einheiten verbundenen äußeren Hardware-Einheiten gehen in den Zustand DETACHED IMPLICITLY, wenn die Verbindung zu der direkt weggeschalteten Einheit die letzte oder die einzige logische Verbindung war. Diese äußeren Hardware-Einheiten können nicht benützt werden.

Das Wegschalten mit dem Parameter SCOPE=*VM2000-GLOBAL ist nur dann erfolgreich, wenn in allen BS2000-Gastsystemen das Wegschalten erfolgreich durchgeführt werden konnte.

Mit dem Operanden FORCE=*UNCONDITIONAL-OFFLINE unter VM2000 wird im Monitorsystem das unbedingte Wegschalten von Kanälen ermöglicht (S-Server).

### INCLUDE-DEVICE-CONNECTION

Die Eingabe eines gültigen Kommandos INCLUDE-DEVICE-CONNECTION hat folgende Wirkung:

Waren die angegebenen logischen Verbindungen im Zustand REMOVED EXPLICITLY, so werden sie INCLUDED. Die Verbindungen können wieder benützt werden.

Waren die zu den Verbindungen gehörenden äußeren Einheiten DETACHED IMPLICITLY, so erhalten sie den Zustand ATTACHED. Diese Hardware-Einheiten können wieder benützt werden.

Die von diesen Hardware-Einheiten ausgehenden äußeren Verbindungen werden INCLUDED, wenn sie vorher REMOVED IMPLICITLY waren. Diese äußeren Verbindungen können wieder benützt werden.

### REMOVE-DEVICE-CONNECTION

Eine gültige Eingabe des Kommandos REMOVE-DEVICE-CONNECTION wirkt sich so aus:

Die angegebenen logischen Verbindungen gehen in den Zustand REMOVED EXPLICITLY. Sie können nicht mehr benützt werden.

Die an diesen Verbindungen angeschlossenen äußeren Hardware-Einheiten erhalten den Zustand DETACHED IMPLICITLY, falls die weggeschaltete Verbindung die letzte oder einzige Verbindung nach innen war. Diese Hardware-Einheiten können nicht mehr benützt werden.

Die von diesen äußeren Hardware-Einheiten im Zustand DETACHED IMPLICITLY ausgehenden äußeren Verbindungen erhalten den Zustand REMOVED IMPLICITLY. Sie können nicht mehr benützt werden.

### Anschlüsse für SKP-Bedienstationen

Die beiden Ein-/Ausgabe-Anschlüsse bilden für das Betriebssystem eine logische Einheit. Bei ATTACH/DETACH-Aufträgen sorgt es dafür, dass beide Anschlüsse den gleichen Zustand einnehmen. Voraussetzung ist, dass auch die beiden Pfade von der Steuerung zu den Anschlüssen im gleichen Zustand sind. Werden mit INCLUDE-/REMOVE-DEVICE-CONNECTION die Pfade in unterschiedliche Zustände gebracht, ist eine ordnungsgemäße Behandlung der Anschlüsse durch das System nicht mehr gewährleistet.



Das Einschalten von Hardware-Einheiten und Verbindungen liegt in der Verantwortung des Operators. Das Ausschalten von Hardware-Einheiten und Verbindungen erfolgt entweder vom Operator oder (bei schwerwiegenden Fehlern) vom System. Der Operator ist verantwortlich für die Koordination des Zu- und Wegschaltens von Hardware-Einheiten, die an mehreren Systemen angeschlossen sind.

Geräte im Zustand DETACH-PENDING können von der Datenträgerüberwachung bei Remount-Aufträgen verwendet werden, falls sonst keine freien Geräte verfügbar sind.

### 5.2.4 Besonderheiten für Magnetband- und Plattengeräte

Bei der System-Initialisierung werden Magnetbandgeräte vom System zunächst explizit DETACHED.

Für ATTACHED-generierte Plattengeräte und Plattengeräte, die zum Home-Pubset gehören, werden über Ein-/Ausgabe-Befehle Geräte- und Steuerungseigenschaften ermittelt. Ein Gerät wird vom System explizit DETACHED, wenn diese Eigenschaften wegen Ein-/Ausgabe-Fehlern nicht ermittelt werden können.

Beim ATTACH eines Magnetband- oder Plattengerätes erfolgt ebenso zunächst die Ermittlung von Geräte- und Steuerungseigenschaften. Ist eine erfolgreiche Ermittlung nicht möglich, wird der ATTACH abgewiesen.

*Ausnahme*

Wurde die Ermittlung der Eigenschaften für Magnetbandgeräte wegen `NO PATH AVAILABLE` (CC=3) abgewiesen, wird der ATTACH zugelassen. Damit ist eine Überkonfigurierung zum Zweck einer Überbelegung möglich.

#### Besonderheiten für MBK-Geräte

Magnetbandkassetten-Geräte (MBK-Geräte) werden nur dann ATTACHED, wenn sie hardwaremäßig für den eigenen Server reserviert werden können.
Nach erfolgreichem Zuschalten sind MBK-Geräte einem Server exklusiv zugeordnet.

*ATTACH-DEVICE*

Beim ATTACH eines MBK-Gerätes wird der Versuch unternommen, das Gerät zu reservieren. Es wird die Meldung `NKR0042 DEVICE=mn  ATTACH ACCEPTED` ausgegeben und das Gerät geht in den Zustand ATTACH-PENDING über.

Nach erfolgreicher Reservierung erfolgt die Meldung `NKR0110 DEVICE=mn  DEVICE ATTACHED AND ASSIGNED` und das MBK-Gerät geht in den Zustand ATTACHED über.

Ist das MBK-Gerät von einem fremden Server aus reserviert, so erfolgen die Meldungen `NKR0111 DEVICE=mn  DEVICE ASSIGNED ELSEWHERE` und `NKR0044 DEVICE=mn  ATTACHMENT REJECTED` und das MBK-Gerät geht in den Zustand DETACHED-EXPLICITLY über.

Kann das MBK-Gerät aus anderen Gründen nicht zugewiesen werden, erfolgt die Meldung `NKR0112 DEVICE=mn  COULD NOT BE ASSIGNED`.

Ein Dummy-MBK-Gerät wird auch dann ATTACHED, wenn die Zuweisung wegen Nichtverfügbarkeit von Gerät oder Gerätesteuerung abgewiesen wird. In diesem Fall werden statt der Meldung `NKR0110` folgende Meldungen ausgegeben:

```
NKR0112 DEVICE=mn  COULD NOT BE ASSIGNED
NKR0040 DEVICE=mn  ATTACHED
```

*DETACH-DEVICE*

Bei DETACH (explizit und implizit) wird versucht, die Hardware-Reservierung zurückzusetzen:

- Während dieser Zeit sind alle vom Rekonfigurationsauftrag betroffenen Geräte/Pfade im Zustand DETACH-/REMOVE-PENDING.
- Nach erfolgreicher Ausführung erfolgt die Meldung `NKR0114 DEVICE=mn  DEVICE UNASSIGNED` oder, falls die Hardware-Reservierung nicht freigegeben werden konnte, die Meldung `NKR0115 DEVICE=mn  COULD NOT BE UNASSIGNED`.
- Ein ATTACH-DEVICE-Kommando während der Wartezeit bewirkt den Versuch einer erneuten Zuweisung.

Auswirkungen eines DETACH-DEVICE/REMOVE-DEVICE-CONNECTION-Kommandos mit dem Operanden FORCE=*YES auf die Steuerung:

1. Die Steuerung quittiert Schreibaufträge, wenn die zu übertragenden Daten in einem Puffer der Steuerung abgelegt sind, und überträgt dann diese Daten asynchron auf den Datenträger.
   Es wird maximal 2 Minuten auf die erfolgreiche Übertragung auf die Kassette bzw. auf das Wiedereinlesen der nicht übertragbaren Daten gewartet. Es wird eine der folgenden Meldungen ausgegeben:
   `NKR0020 DETACH DEVICE ACCEPTED`
   `NKR0021 REMOVE DEVICE CONNECTION ACCEPTED`

   Während dieser Wartezeit befindet sich das Kassettengerät im Zustand DETACH-PENDING.
2. Spätestens nach Ablauf dieser Wartezeit wird der Versuch unternommen, die Hardware-Reservierung freizugeben.
3. Ein ATTACH-DEVICE-Kommando während der Wartezeit bewirkt ein Aufheben des DETACH-Auftrags.

Bei Systembeendigung wird der Versuch unternommen, alle (noch) zum Server bestehenden Hardware-Reservierungen freizugeben. Ein Scheitern der Freigabe wird durch folgende Meldung angezeigt:
`NKR0115 DEVICE=mn  COULD NOT BE UNASSIGNED`

Wird ein Zuschaltversuch mit der Meldung `NKR0111 DEVICE=mn  DEVICE ASSIGNED ELSEWHERE` abgewiesen, so ist das Gerät von einem fremden Server reserviert. Mit dem Kommando UNLOCK-DEVICE kann dem fremden Server die Reservierung entzogen werden. Nach einem Systemabsturz kann mit diesem Kommando das Gerät freigeschaltet werden.

## 5.3 Dynamische I/O-Konfigurationsänderung (S-Server)

Rekonfiguration im laufenden Betrieb ist das Zu- und Wegschalten bereits definierter Komponenten der Server-Konfiguration (siehe Abschnitt „Rekonfiguration“ auf Seite 197).

An S-Servern besteht darüber hinaus die Möglichkeit, Ein-/Ausgabe-Einheiten während des laufenden Betriebs der Server-Konfiguration hinzuzufügen bzw. daraus zu entfernen.

An den S-Servern kann die Ein-/Ausgabe-Konfiguration im laufenden Betrieb geändert werden, d.h. die bestehende Konfiguration kann ohne neuen IMPL erweitert oder reduziert werden. Die Änderungen werden direkt auf dem aktiven IORSF des Servers ausgeführt. Da diese Server mit Typ-S- oder Typ-FC-Kanälen ausgestattet sein können und eine Neu- bzw. Umverkabelung von Ein-/Ausgabe-Komponenten am Glasfaserkabel einfach und störungsfrei durchführbar ist, bieten diese Server ideale Voraussetzungen für die unterbrechungsfreie Änderung der Ein-/Ausgabe-Konfiguration.

Diese Möglichkeit der Konfigurationsänderung wird im Folgenden als „dynamische I/O-Konfigurationsänderung“ bezeichnet. Die dynamische I/O-Konfigurationsänderung wird für folgende Geräteklassen unterstützt:

- Platten- und Bandgeräte
- Drucker
- DFÜ-Geräte
- Exotengeräte

#### Vorbereitung der dynamischen I/O-Rekonfiguration

Damit im laufenden Betrieb neue Geräte und Steuerungen der Konfiguration hinzugefügt werden können, erzeugt das System bei der Systemeinleitung in der Gerätetabelle zusätzliche freie Tabelleneinträge für Geräte und Steuerungen. Die Tabelleneinträge für Kanäle und Verbindungen zwischen Ein-/Ausgabe-Einheiten werden dynamisch angelegt und müssen von der Systembetreuung bei der Generierung des IORSF nicht mit berücksichtigt werden.

Die Anzahl der freien Tabelleneinträge für Geräte und Steuerungen richtet sich nach der Größe des Reservebereichs im IORSF.
Dieser reicht bei einem mit IOGEN erstellten IORSF für 512 zusätzliche Einträge (Kanäle, Steuerungen, Geräte und deren Verbindungen), siehe Handbuch „Systeminstallation“ [57].
Ein bereits modifiziertes IORSF kann weniger oder mehr freie Einträge besitzen, entweder durch Hinzufügen oder Wegnahme von Einheiten im aktiven IORSF oder durch Zurückschreiben des aktiven IORSF mit dem Kommando STOP-CONFIGURATION-UPDATE ..., DEVICE-RESERVE=.

Bei der Systemeinleitung werden für Drucker, DFÜ-Geräte und Exotengeräte zusammen 64 freie Einträge in der Gerätetabelle vorgesehen, für Platten- und Bandgeräte jeweils die aufgrund der freien Einträge im IORSF maximal mögliche Anzahl (höchstens aber je 2048). Die Gesamtzahl der dynamisch hinzufügbaren Einheiten kann dabei nicht die im IORSF gesetzten Grenzen überschreiten.

*Maßnahmen zur Vermeidung von Engpässen*

Jedes Gerät oder jede Steuerung, die der Konfiguration neu hinzugefügt wird, belegt einen freien Tabelleneintrag. Das Hinzufügen wird abgewiesen, wenn kein freier Tabelleneintrag mehr existiert. In diesem Fall ist ein Hinzufügen erst wieder möglich, wenn eine entsprechende Ein-/Ausgabe-Einheit entfernt wird.

Ist bereits bei der Generierung des IORSF abzusehen, dass diese freien Tabelleneinträge für die in der nächsten Session vorgesehene dynamische I/O-Konfigurationsänderung nicht ausreichen werden, sollte eine genügend große Anzahl zukünftiger Ein-/Ausgabe-Einheiten im IORSF mitkonfiguriert werden. Diese zusätzlichen Ein-/Ausgabe-Einheiten können dann zum Zeitpunkt der Inbetriebnahme mit den Kommandos der dynamischen Rekonfiguration entsprechend der realen Konfiguration umdefiniert werden.

### Dynamische I/O-Konfigurationsänderung vornehmen

Folgende Kommandos stehen zur dynamischen I/O-Konfigurationsänderung zur Verfügung. Die Kommandos sind im Handbuch „Kommandos“ [27] ausführlich beschrieben:

| **Kommando** | **Bedeutung** |
|---|---|
| ADD-IO-UNIT | neue Ein-/Ausgabe-Einheiten definieren |
| MODIFY-IO-UNIT | Pfadbeschreibung einer Steuerung ändern,<br>bevorzugten Pfad eines Gerätes ändern,<br>Timeout-Einstellungen ändern,<br>bevorzugtes Gerät für PAV festlegen |
| REMOVE-IO-UNIT | Ein-/Ausgabe-Einheiten entfernen |
| SHOW-DEVICE-CONFIGURATION | Abfragen der Ein-/Ausgabe-Konfiguration |
| START-CONFIGURATION-UPDATE | dynamische I/O-Konfigurationsänderung einleiten |
| STOP-CONFIGURATION-UPDATE | dynamische I/O-Konfigurationsänderung beenden,<br>aktuelles IORSF sichern |

Tabelle 14: Komandos der dynamischen I/O-Konfigurationsänderung

Bevor Änderungen an der Konfiguration vorgenommen werden, muss die dynamische I/O-Konfigurationsänderung mit dem Kommando START-CONFIGURATION-UPDATE eingeleitet werden. Die Kommandos zur Änderung der Konfiguration werden erst nach erfolgreicher Einleitung angenommen.

*Ausnahme*

Sollen nur die Timeout-Einstellungen von Geräten verändert werden, kann das Kommando MODIFY-IO-UNIT (mit UNIT=*DEVICE(...) und TIMEOUT=...) auch außerhalb einer dynamischen Rekonfiguration eingegeben werden. Dasselbe gilt für die Festlegung eines bevorzugten Gerätes für PAV (Operand PAV-PREFERRED-DEVICE). **Diese Änderungen sind unabhängig vom Server-Typ möglich!**

Ist die dynamische I/O-Konfigurationsänderung erfolgreich eingeleitet, kann die Systembetreuung im laufenden System mit den Kommandos ADD-IO-UNIT, REMOVE-IO-UNIT und MODIFY-IO-UNIT die Ein-/Ausgabe-Konfiguration dynamisch ändern. Die Änderungen werden im aktiven IORSF (Input/Output Resource File) vorgenommen, d.h. sie werden sofort wirksam.

### Allgemeiner Ablauf einer dynamischen I/O-Konfigurationsänderung

Jede über Kommando angeforderte Konfigurationsänderung wird mehrstufig unter Steuerung der Systemkomponente IORECONF abgearbeitet:

1. Prüfphase:
   IORECONF prüft die Durchführbarkeit der angeforderten Konfigurationsänderung und weist nicht durchführbare Anforderungen zurück. So wird z.B. ein ADD-IO-UNIT-Kommando für ein Gerät aus einem der folgenden Gründe abgewiesen:
   - Die Steuerung oder der Kanal, an dem das Gerät angeschlossen werden soll, ist nicht definiert.
   - Das Gerät ist bereits definiert.
   - Es ist kein freier Tabelleneintrag vorhanden.

   Danach werden alle weiteren Systemkomponenten, die sich für Konfigurationsänderungen „interessieren“, d.h. die sich zuvor bei IORECONF angemeldet haben, aufgerufen (z.B. die Komponenten der Geräte- und Datenträgerverwaltung). Die Anforderung kann dann durch eine der gerufenen Komponenten abgelehnt werden.

2. Modifikation des IORSF:
   Nach abgeschlossener Prüfung wird die Konfigurationsänderung in das aktive IORSF eingetragen. Für die Dauer der IORSF-Änderung sind alle betroffenen Geräte für Ein-/Ausgaben gesperrt.

3. Modifikation der Tabellen:
   Nach erfolgreicher IORSF-Änderung ändert IORECONF die Ein-/Ausgabe-Tabellen. IORECONF informiert die angemeldeten Systemkomponenten über die Konfigurationsänderung, damit diese ihre Tabellen ebenfalls aktualisieren können.

Kann die Anforderung einer Konfigurationsänderung nicht vollständig ausgeführt werden, versucht IORECONF den Ausgangszustand wiederherzustellen.
Sowohl Durchführung als auch Abweisung von Anforderungen zur Konfigurationsänderung werden durch Konsol-Meldungen (`NKR....`) protokolliert.

### I/O-Konfiguration dynamisch erweitern

Mit dem Kommando ADD-IO-UNIT kann die Systembetreuung der Konfiguration neue Ein-/Ausgabe-Einheiten (Geräte, Steuerungen, Kanäle) hinzufügen. Geräte können als PAV-Alias-Geräte definiert werden (siehe „PAV am FC-Kanal“ auf Seite 247).

Neue Ein-/Ausgabe-Einheiten müssen in der Reihenfolge Kanal - Steuerung - Gerät angegeben werden. Bis zu 256 Geräte lassen sich mit einem ADD-IO-UNIT-Kommando an einer Steuerung hinzufügen.

*Einschränkungen und Besonderheiten*

Die uneingeschränkte Funktionalität ist nur für folgende Kanäle sowie die daran angeschlossenen Geräte und Gerätesteuerungen gegeben:

- Kanal Typ S (Kanäle vom IOGEN-Kanaltyp IBS mit MODE=CVC/CNC)
- Kanal Typ FC (Kanäle vom IOGEN-Kanaltyp IBF mit MODE=FCP) [1]

Kanäle vom Typ IBS und MODE CTC sowie andere Kanaltypen als IBS und IBF werden nicht unterstützt. An CTC-Kanälen ist die Rekonfiguration von Steuerungen und Geräten möglich, an Kanälen eines Typs ungleich IBS nur die Rekonfiguration von Geräten.

Für die Steuerung von Kanälen vom Typ IBS oder IBF können logische Adressen vereinbart werden, weil diese beiden Kanaltypen die Bildung logischer Steuerungen zulassen.
Mit dem Operanden LOGICAL-ADDRESS können angegeben werden:

- für Kanal Typ S (MODE=CVC) logische Adressen zwischen 0 und 15
- für Kanal Typ FC (MODE=FCP) logische Adressen zwischen 0 bis 255

Eine neue Kanalbaugruppe kann mit folgenden Schritten hinzugefügt werden:

1. Kommando ADD-IO-UNIT *CHN(...),STATE=DET für jeden Kanal der Baugruppe
2. Einbau der Kanalgruppe und Aktivierung über CM-Frame
3. Kommando ATTACH-DEVICE *CHN(...) für jeden Kanal der Baugruppe

Kanäle auf bereits bei IMPL vorhandenen Baugruppen können zwar mit ADD-IO-UNIT dynamisch generiert werden, sie lassen sich danach aber nicht „online“ schalten. Deshalb sollten auch alle bereits vorhandenen Kanäle vorher generiert werden.

[1] FC (Fibre Channel) ist der Standard für Server-Storage-Verbindungen in der Open Systems Welt. Auf S-Servern wird er durch den Kanal Typ FC realisiert.

Für den Betrieb eines neuen Bandgerätes sind nach der Neudefinition mit ADD-IO-UNIT und bei Einsatz eines Roboters noch folgende Maßnahmen erforderlich:

1. Mit dem Kommando ADD-DEVICE-DEPOT muss eine Lagerortzuordnung getroffen werden.
2. Falls MAREN im Einsatz ist und das neue Gerät der Freibandzuweisung unterliegen soll, muss das Programm MARENUCP beendet und neu gestartet werden.
3. Ist das neue Gerät Teil eines von ROBAR gesteuerten Tape-Library-Systems und noch nicht in der ROBAR-Konfigurationsdatei definiert, muss ROBAR-SV beendet, die ROBAR-SV-Konfigurationsdatei erweitert und daraus eine Binärdatei erstellt werden. Danach kann ROBAR-SV neu gestartet werden.

Bei Konfiguationen ohne Roboter muss das Bandgerät nur noch ATTACHED werden (Kommando ATTACH-DEVICE), wenn es nicht bereits implizit mit dem Kommando ADD-IO-UNIT ...,STATE=*ATTACHED zugeschaltet wurde.

Ein neuer Drucker muss vor oder nach dem Hinzufügen in die SPOOL-Parameterdatei aufgenommen werden (SPSERVE-Anweisung ADD-SPOOL-DEVICE).

Ein neues DFÜ-Gerät muss vor oder nach dem Hinzufügen mit BCIN-Kommando im DCM bekannt gemacht werden.

Das Subsystem IOTRACE berücksichtigt neu hinzugefügte Ein-/Ausgabe-Einheiten nur, wenn es nach der Konfigurationsänderung neu gestartet wurde.

### I/O-Konfiguration dynamisch verkleinern

Mit dem Kommando REMOVE-IO-UNIT kann die Systembetreuung Ein-/Ausgabe-Einheiten (Geräte, Steuerungen, Kanäle), die sich im Zustand DETACHED befinden, aus der aktuellen Konfiguration entfernen. Wird eine „innere“ Einheit entfernt (Kanal bzw. Steuerung), so werden vom System implizit die zugehörigen „äußeren“ Einheiten (Steuerung bzw. Geräte) entfernt, sofern diese keiner weiteren „inneren“ Einheit angehören. Zu entfernende Ein-/Ausgabe-Einheiten müssen in der Reihenfolge Gerät - Steuerung - Kanal angegeben werden.

### I/O-Konfigurationseigenschaften dynamisch modifizieren

Mit dem Kommando MODIFY-IO-UNIT kann die Systembetreuung die Konfigurationseigenschaften von Steuerungen oder Geräten dynamisch verändern. Bei einer Steuerung können die Ein-/Ausgabe-Pfade und die Verfügbarkeit der Verbindung zwischen Kanal und Steuerung (INCLUDED oder REMOVED) neu festgelegt werden.

Somit kann eine Steuerung, die mindestens zwei Kanalanschlüsse besitzt, unterbrechungsfrei umgehängt werden. Bei einem Gerät kann der bevorzugte Ein-/Ausgabe-Pfad geändert werden. Mit dem Operanden STATE=*PAV-PREFERRED-DEVICE(...) kann festgelegt werden, dass das PAV-Gerät unter VM2000 bei Ein-/Ausgaben zu bevorzugen ist (zu PAV siehe Abschnitt „PAV: Parallel Access Volume (S-Server)“ auf Seite 244).

Außerdem kann mit dem Kommando die Timeout-Einstellung von Geräten verändert werden (MODIFY-IO-UNIT UNIT=*DEVICE(...),TIMEOUT=<wert_in_sekunden>).
Werden nur die Timeout-Einstellungen von Geräten verändert oder ein bevorzugtes Gerät für PAV festgelegt, kann das Kommando auch außerhalb einer dynamischen Rekonfiguration (d.h. auch unabhängig vom Server-Typ) eingegeben werden.
Die Veränderung der Timeout-Werte sollte nur kurzfristig für bestimmte Aktionen, z.B. den Online-Update einer Plattenspeichersystem-Steuerung, erfolgen. Danach sollten wieder die Standardwerte mit TIMEOUT=*DEFAULT eingestellt werden.

### Dynamische I/O-Konfigurationsänderung beenden

Die dynamische I/O-Konfigurationsänderung wird mit dem Kommando STOP-CONFIGURATION-UPDATE abgeschlossen. Damit die aktuelle Ein-/Ausgabe-Konfiguration auch für eine Folge-Session verfügbar ist, kann das aktuelle IORSF über den Operanden IORSF-UPDATE auf die SVP-Hard-Disk zurückgeschrieben werden.
Kommt es vor dem Zurückschreiben des aktiven IORSF zu einem Systemabsturz, gehen die bis dahin vorgenommenen Konfigurationsänderungen beim erneuten IPL nicht verloren, da die BS2000-Ein-/Ausgabe-Tabellen bei der Systemeinleitung aus dem aktiven IORSF abgeleitet werden.

Nach erfolgreichem Abschluss der dynamischen Rekonfiguration werden die Rekonfigurationskommandos ADD-IO-UNIT, REMOVE-IO-UNIT und MODIFY-IO-UNIT (Ausnahme Timeout-Einstellung) nicht mehr angenommen.

### Dynamische I/O-Konfigurationsänderung unter VM2000

Unter VM2000 können die Kommandos zur dynamischen I/O-Rekonfiguration nur im Monitor-System eingegeben werden. Die erforderlichen Änderungen der Ein-/Ausgabe-Tabellen werden im Hypervisor, im Monitorsystem und automatisch in allen aktiven Gastsystemen durchgeführt. Bei Eingabe des START-CONFIGURATION-UPDATE-Kommandos wird in der Meldung `NKR0178` angezeigt, wie viele Gastsysteme aktiv sind und die dynamische I/O-Konfigurationsänderung unterstützen.

## 5.4 Dynamische I/O-Konfigurationsänderung (SQ-Server)

An SQ-Servern wird Busperipherie nicht mit IOGEN, sondern über X2000 konfiguriert. Damit entfällt auch die Möglichkeit der Überdefinition der Ein-/Ausgabe-Konfiguration, die es erlaubt, vordefinierte Geräte zu einem späteren Zeitpunkt dynamisch in Betrieb zu nehmen.

Bus-Platten, Band- und DFÜ-Geräte sowie Drucker können im laufenden Betrieb der Konfiguration hinzugefügt werden. BS2000/OSD erzeugt für diesen Zweck bei der Systemeinleitung zusätzliche freie Gerätetabelleneinträge:

- für DFÜ-Geräte und Drucker zusammen 64
- für Platten- und Bandgeräte jeweils ca. 10% der Anzahl bereits in der Gerätetabelle definierten Geräte (mindestens 64, höchstens 2048)

## 5.5 Rekonfiguration von Extra- und Spare-CPUs

S-Server bieten gesteigerte Hochverfügbarkeit für CPUs sowie die Möglichkeit, über die nominelle Server-Leistung hinaus zeitlich begrenzt weitere, bereits vorinstallierte CPUs zuzuschalten.

Bei Ausfall einer CPU schaltet BS2000/OSD automatisch eine vorhandene Ersatz-CPU (Spare-CPU) zu, womit die nominelle Leistung des Servers garantiert bleibt.

Bei Bedarf an zeitlich begrenzter zusätzlicher Leistung des Servers können über den nominellen Multiprozessorgrad (MP-Grad) des Servers hinaus neben der Spare-CPU weitere CPUs (Extra-CPUs) installiert sein, deren Nutzungsdauer vertraglich geregelt ist. Das Zu- und Wegschalten erfolgt durch den Systembetreiber mittels erweiterter CPU-Rekonfigurationskommandos. Die Nutzungsintervalle werden in BS2000/OSD in einer eigenen Protokolldatei sowie als Meldungen an die Teleservice-Zentrale aufgezeichnet.

Zur Nutzung von Spare-/Extra-CPUs in VM2000 siehe Handbuch „VM2000“ [62].

i Auf SQ-Servern werden im Rahmen der „Capacity on Demand“-Konzepte ebenfalls Extra-CPUs angeboten.

### 5.5.1 Funktionalität im native BS2000-Betrieb

**Installation**

Im Rahmen der Neuinstallation bzw. der Umkonfiguration eines BS2000/OSD-Servers hinterlegt die vom Service eingelesene Modell-Lock-Diskette die Modell- und HW-Charakteristika im Server. Das sind: die Modell-ID, die global eindeutigen IDs der installierten CPUs, der nominelle MP-Grad des Servers sowie die Anzahl der verfügbaren Spare-CPUs (nur bei S-Servern) sowie der vertraglich vereinbarten Extra-CPUs.

**IPL**

Beim Startup werden gemäß dem nominellen MP-Grad CPUs in Betrieb genommen (ATTACHED, ONLINE, NORMAL).

Extra-CPUs bleiben in BS2000/OSD weggeschaltet (DETACHED, OFFLINE, EXTRA).

Spare-CPUs bleiben ebenfalls weggeschaltet (DETACHED, OFFLINE, SPARE), falls es sich um ein Modell mit nominell zwei oder mehr CPUs handelt. Bei Modellen mit einer nominellen CPU wird die Spare-CPU in den Zustand DETACHED, ONLINE, SPARE gebracht.

Die Festlegung einer CPU für den Normalbetrieb erfolgt jeweils bei IPL durch die Firmware und kontrolliert durch die BS2000-Software; der Zustand der CPUs wird mit dem Kommando SHOW-DEVICE-CONFIGURATION CLASS=*CPU angezeigt.

**CPU-Ausfall**

Bei Ausfall einer CPU (MFA=Malfunction Alert, MCKI=Machine Check Interrupt) auf einem S-Server schaltet BS2000/OSD die defekte CPU (DETACHED, OFFLINE, ERROR) weg und gleichzeitig eine betriebsbereite Spare-CPU zu.

Dabei werden folgenden Meldungen, die auch als Teleservice-Call an das zuständige Service-Center gemeldet werden, ausgegeben:

```
ETMRK48 CPU (&01) DETACHED BY SYSTEM
ETMRK20 CPU (&01) ATTACHED BY SYSTEM
```

Die weggeschaltete CPU ist nach einem MFA für folgende „IPL/Power on Reset“ bzw. IMPL gesperrt. BS2000/OSD zeigt sie ab dem nächsten IPL bzw. IMPL nicht mehr an. Erst nach Reparatur bzw. Austausch durch den Service (der ganze Server muss hierzu offline sein) kann dieser die CPU per SVP-Frame wieder verfügbar erklären.

Beim CPU-Ausfall aus anderen Gründen (z.B. CPU Loop) findet das Wegschalten der CPU herkömmlich ohne Auswirkung für das nächste Hochfahren statt (DETACHED, ONLINE, NORMAL). Das ersatzweise Zuschalten einer Spare-CPU findet nicht statt.

### Zu- und Wegschalten von Extra-CPUs

Gemäß der vertraglich vereinbarten Nutzungsdauer können Extra-CPUs für eine begrenzte Anzahl von Tagen vom Betreiber zugeschaltet werden. Im BS2000-Betrieb erfolgt dies mit erweiterten Operanden der Kommandos ATTACH-DEVICE bzw. DETACH-DEVICE:

```
/ATTACH-DEVICE UNIT=*EXTRA-CPU(CPU-IDENTIFIER=<x-string 2..2>/*ALL/*ANY)
```

mit:

*ALL Alle weggeschalteten Extra-CPUs sind zuzuschalten.
*ANY Eine beliebige der weggeschalteten Extra CPUs ist zuzuschalten.

Zusätzliche Meldungen in der Bearbeitung des Kommandos (asynchron, keine Kommando-Returncodes):

```
ETMRK58 EXTRA-CPU (&00) ATTACHED
ETMRK5A ALL EXTRA-CPU'S ALREADY ATTACHED
ETMRK5E OBJECT EXTRA-CPU NOT EXISTING ON THIS HARDWARE
```

Das Wegschalten der Extra-CPUs erfolgt analog:

```
/DETACH-DEVICE UNIT=*EXTRA-CPU(CPU-IDENTIFIER=<x-string 2..2>/*ALL/*ANY)
```

mit:

*ALL Alle zugeschalteten Extra-CPUs sind wegzuschalten.
*ANY Eine beliebige der zugeschalteten Extra CPUs ist wegzuschalten.

Zusätzliche Meldungen in der Bearbeitung des Kommandos (asynchron, keine Kommando-Returncodes):

```
ETMRK59 EXTRA-CPU (&00) DETACHED
ETMRK5B ALL EXTRA-CPU'S ALREADY DETACHED
ETMRK5E OBJECT EXTRA-CPU NOT EXISTING ON THIS HARDWARE
```

Das Zu- und Wegschalten wird in der speziellen Protokolldatei SYS.RESLOG festgehalten (siehe „Diagnosehandbuch“ [14]).

Der Betreiber ist verpflichtet, eine derartige (Jahres-)Auswertung an den zuständige Vertriebsbeauftragten zu schicken. Zusätzlich werden die Ereignisse „Zuschalten“ bzw. „Wegschalten“ an die Teleservice-Zentrale übertragen (Meldungen ETMRK58 bzw. ETMRK59).

Auch bei Ausfall einer Extra-CPU wird eine betriebsbereite Spare-CPU automatisch zugeschaltet (nur auf S-Servern).

Die Nutzung von Extra-Leistung endet mit Ende der BS2000-Session bzw. bei deren abnormaler Beendigung. Mit Beginn einer neuen Session sind Extra-CPUs wieder explizit hinzuzunehmen.

### Erweiterung der Ausgabe in Informationskommandos

*Beispiel: Informationen über CPUs (S-Server mit 2 Extra- und 1 Spare-CPU*

```
/SHOW-DEVICE-CONFIGURATION UNIT=*SELECT(CLASS=*CENTRAL-PROC)

MNEM  UN-CLASS  UN-TYPE  CONF-STATE  POOL/SIDE

00    CPU       S190-30  ATTACH /ON  NORMAL
01    CPU       S190-30  ATTACH /ON  NORMAL
02    CPU       S190-30  ATTACH /ON  NORMAL
03    CPU       S190-30  ATTACH /ON  EXTRA
04    CPU       S190-30  DET(EX)/OFF EXTRA
05    CPU       S190-30  DET(EX)/OFF SPARE
```

## 5.5.2 Funktionalität im Gastsystem unter VM2000

BS2000-Gastsystemen sind die Extra-CPUs verborgen. Ihre Nutzung ist ausschließlich über VM2000-Kommandos möglich, siehe Handbuch „VM2000“ [62].

In VM2000 werden allen Gastsystemen neben den normalen virtuellen CPUs (deren Anzahl beim Einrichten der VM festgelegt wird) so viele virtuelle Spare-CPUs zugewiesen, wie der Server reale Spare-CPUs konfiguriert hat. Die bei Ausfall einer realen CPU momentan darauf ablaufende und damit ebenfalls ausfallende virtuelle CPU wird vom betroffenen Gastsystem durch eine virtuelle Spare-CPU ersetzt.

Dies ist von besonderer Bedeutung für die Verfügbarkeit von Gastsystemen, denen nur eine CPU zugewiesen wurde, da solche Systeme ohne Spare-CPU beim Auftreten von CPU-Fehlern während ihres Ablaufs sofort beendet sind. Mit einer (virtuellen) Spare-CPU gibt es im Fehlerfall eine realistische Chance zum unterbrechungsfreien Weiterlauf des Gastsystems nach interner CPU-Rekonfiguration.

## 5.6 NDM – Betriebsmittelbelegung und -reservierung

Das NDM (Nucleus Device Management) ist die eigentliche Geräteverwaltung von BS2000/OSD. Grundfunktion der Geräteverwaltung ist die Gerätebelegung. Da i.A. nicht immer jede Geräteanforderung erfüllbar ist, muss NDM auch eine Gerätereservierungsfunktion anbieten sowie Warteschlangen verwalten und abarbeiten.

NDM erfasst Anzahl und Zustand der Geräte beim Systemstart sowie Zustandsänderungen während des laufenden Betriebs.

Stellt ein Benutzerprogramm einen Antrag auf ein bestimmtes Gerät, ist dieses verfügbar und wird ihm von NDM zugewiesen, so muss NDM auch dafür sorgen, dass das Gerät als belegt gekennzeichnet wird und kein anderes Benutzerprogramm Zugang zu dem Gerät hat. Dieses Gerät gilt als belegt, bis das zugriffsberechtigte Benutzerprogramm seine Arbeit mit dem Gerät beendet und dies NDM mitgeteilt hat.

| Kommando | Bedeutung |
|---|---|
| ADD-DEVICE-DEPOT | Zuordnung Bandgerät zu Lagerort vereinbaren |
| CHANGE-DISK-MOUNT | Privatplatte für Zugriff sperren |
| CHANGE-TAPE-MOUNT | Montierzustand von Bändern ändern |
| CHECK-DISK-MOUNT | Montierzustand einer Platte prüfen |
| CHECK-TAPE-MOUNT | Montierzustand von Bandgeräten und MBK prüfen |
| MODIFY-MOUNT-PARAMETERS | Vorgaben für das Montieren und Demontieren von Bändern und Platten ändern |
| MODIFY-RESOURCE-COLLECTION | Collector-Auswahl steuern |
| REMOVE-DEVICE-DEPOT | Zuordnung Bandgerät zu Lagerort aufheben |
| SECURE-RESOURCE-ALLOCATION | Fordert Betriebsmittel an |
| SET-DISK-DEFAULTS | Standardwerte für DISK-Parameter festlegen |
| SET-DISK-PARAMETER | Vorgaben für die Überwachung von Platten setzen |
| SET-DRV-PARAMETER [1] | Aufzeichnungsverfahren für Privatplatte bzw. Pubset festlegen |
| SHOW-DEVICE-CONFIGURATION | Abfragen der Ein-/Ausgabe-Konfiguration |
| SHOW-DEVICE-DEPOT | Zuordnung Bandgerät zu Lagerort abfragen |
| SHOW-DEVICE-STATUS | Belegungs- und Überwachungsinformationen für Geräte anfordern |
| SHOW-DISK-STATUS | Belegung und DISK-Parameter abfragen |
| SHOW-DRV-STATUS [1] | DRV-spezifische Informationen und Vorgaben der Parameter erfragen |

Tabelle 15: Schnittstellenübersicht zur Betriebsmittelbelegung und -reservierung (Teil 1 von 2)

| **Kommando** | **Bedeutung** |
|---|---|
| SHOW-MOUNT-PARAMETER | Monitor-Vorgaben von Platten und Bändern abfragen |
| SHOW-RESOURCE-ALLOCATION | Task-Belegung und offene Operator-Aktionen abfragen |
| SHOW-RESOURCE-REQUESTS | Zustand von Gerätewarteschlange und Collector-Task abfragen |
| SHOW-TAPE-STATUS | Bandbelegung und -überwachung abfragen |
| START-DRV-DUAL-MODE [1] | Doppelführung der Daten im DUAL-Modus starten |
| START-RESOURCE-COLLECTION | Collector-Auswahl starten |
| STOP-DRV-DUAL-MODE [1] | DUAL-Modus für ein Plattenpaar rücksetzen |
| STOP-RESOURCE-COLLECTION | Collector-Auswahl beenden |
| UNLOCK-DISK | Systembelegungs-Protokoll bereinigen |
| **Makro** | **Bedeutung** |
| NKDINF | Information über den Belegungs- und Verfügbarkeitszustand der (peripheren) Konfiguration |
| NKGTYPE | Information über Namen, Gerätetypcode, Geräteeigenschaften usw. eines Geräte- oder Volumtyps oder über Namen und Gerätetypcodes der Gerätetypen, die zu einer Gerätefamilie oder -klasse gehören |

Tabelle 15: Schnittstellenübersicht zur Betriebsmittelbelegung und -reservierung (Teil 2 von 2)

[1] Diese Kommandos stehen der Systembetreuung nur mit dem Einsatz des Produktes DRV zur Verfügung

Die NDM-Informationsdienste liefern dem Operator bestimmte, dem angegebenen Kommando und dem gewünschten Umfang der Information entsprechende Ausgabefelder.

Eine Übersicht über die SHOW-Kommandos und die Erklärung der ausgegebenen Informationen finden Sie im Anhang des Handbuchs „Kommandos“ [27].

### 5.6.1 Task-Belegung von Datenträgern

Die Geräteverwaltung steuert und überwacht die Belegung der vom Benutzer angeforderten Betriebsmittel (Geräte, Datenträger, Dateien), abhängig von der Benutzungsart der Datenträger.

Für Bänder und private Platten gibt es die Benutzungsarten:

- „DMS“
  für alle DVS-Zugriffe (über Kommandos SECURE-RESOURCE-ALLOCATION, CREATE-FILE, ADD-FILE-LINK, COPY-FILE...)
- „SPECIAL“
  für Sonderanwendungen (z.B. Dienstprogramme INIT, VOLIN, FDDRL ...)

Für Bänder gibt es darüber hinaus noch die Benutzungsart:

- „WORK“
  für DVS-Arbeitsbänder, siehe Kommandos SECURE-RESOURCE-ALLOCATION, ADD-FILE-LINK im Handbuch „Kommandos“ [27]

#### Task-exklusive Belegung

Ein Datenträger kann nicht gleichzeitig von mehreren Tasks belegt werden.
Alle Belegungsanforderungen anderer Tasks für denselben Datenträger innerhalb desselben Systems sowie auch von anderen Systemen werden abgewiesen.

Ausnahme: Exklusive Belegungsanforderungen (SECURE-RESOURCE-Kommando) von Batch-Tasks warten, bis der Datenträger freigegeben wird.

#### Task-shareable Belegung

Ein Datenträger kann gleichzeitig von mehreren Tasks belegt werden. Die einzigen Datenträger, die task-shareable belegt werden können, sind Platten im USE-MODE=DMS (gemeinschaftliche und Privatplatten).

**Belegungsarten für private Datenträger und sonstige Betriebsmittel:**

- Belegung von Bändern

  Bänder werden immer task-exklusiv belegt, eine task-shareable Belegung ist nicht möglich. Das System unterscheidet, ob das Band nur belegt oder tatsächlich genutzt werden soll. Wird es nur belegt, d.h. andere Tasks können nicht auf das Band zugreifen, es finden jedoch keine I/O-Operationen statt, so wird der Operator mit einer Premount-Meldung aufgefordert, eine Gerätezuordnung zu treffen, falls das Band nicht als montiert erkannt wird. Soll das Band dagegen für I/O-Operationen genutzt werden, wird der Operator mit einer Mount-Aufforderung auf diese Absicht hingewiesen.

- Belegung von Privatplatten

  Privatplatten werden für DVS-Betrieb (Benutzungsart DMS) standardmäßig task-shareable belegt. Task-exklusive Belegung ist nur über eine entsprechende Vorab-Reservierung (SECURE) möglich.

  Platten für Benutzungsart SPECIAL werden task-exklusiv belegt.

  Der Operator kann mit dem DISK-Parameter USER-ALLOCATION festlegen, welche Task-Belegung für eine Platte (für Benutzungsart DMS) zugelassen ist.

  Mit USER=*EXCL kann eine Platte nur task-exklusiv (mit SECURE-RESOURCE-ALLOCATION DISK= ...,ALLOC=*EXCL) belegt werden. Die auf diese Weise belegte Platte ist dann nur für diese Task verfügbar. Jeder Versuch, diese Platte task-shareable zu belegen, wird abgewiesen.

  Mit USER=*SHARE kann die Platte für alle DVS- und SECURE-Anforderungen ausschließlich task-shareable belegt werden. Jede task-exklusive Belegungsanforderung der Platte wird abgewiesen.

  Mit USER=*ALL ist die Platte für alle task-shareablen und -exklusiven DVS- und Secure-Anforderungen belegbar. Dies gilt jedoch nur für die Erstbelegung der Platte. Wird die Platte task-exklusiv belegt, werden alle weiteren Belegungsanforderungen von anderen Tasks für diese Platte abgewiesen.

  Mit USER=*NO kann die Platte von keinem Benutzer belegt werden. Diese Einstellung erfolgt vom System automatisch bei: `/CHANGE-DISK-MOUNT UNIT=mn/*VOLUME(vsn), ACTION=*CANCEL` oder bei Beantwortung einer Remount-Meldung mit `tsn.N`.

- Belegung von sonstigen Betriebsmitteln

  Alle Belegungsanforderungen für Geräte (Drucker, Band-, Plattengeräte, etc) werden ohne Eingriff durch den Operator task-exklusiv durchgeführt.

### 5.6.2 Systembelegung von Platten

Jede Belegung einer Platte wird in ihrem Standard-Volume-Label (SVL) hinterlegt. Dazu beinhaltet das SVL der Platte ein Systembelegungs-Protokoll, in das bei Privatplatten bis zu 4 und bei gemeinschaftlichen Platten bis zu 16 System-Katalogkennungen, die die Platte belegen, eingetragen werden können. Für Platten gibt es folgende Belegungen:

**System-exklusive Belegung:**

Eine Platte kann nur von einem System belegt werden. Die System-Id des Home-Pubsets dieses Systems wird im Systembelegungs-Protokoll des SVL so eingetragen, dass dies eine gleichzeitige Belegung der Platte durch andere Systeme verhindert.

**System-shareable Belegung:**
(Einsatz einer Privatplatte im SPD-Betrieb)

Eine Privatplatte kann von maximal 4 Systemen gleichzeitig belegt werden (SPD-Betrieb). Die System-Id jedes Systems, das die Platte belegt, wird so im Systembelegungs-Protokoll eingetragen, dass bis zu max. 3 weitere Systeme gleichzeitig die Platte belegen können. System-shareable und task-exklusive Belegung einer Platte schließen sich gegenseitig aus. Damit eine system-shareable belegte Platte von mehreren Tasks (aus dem gleichen System oder aus verschiedenen Systemen) benutzt werden kann, muss bei der Einstellung des DISK-Parameters SYSTEM-ALLOCATION=*SHARE auch der Parameter USER-ALLOCATION=*SHARE/*ALL gesetzt sein (Kommando SET-DISK-PARAMETER).

Plattengeräte werden immer shareable generiert. Privatplatten sind damit also auch immer SPD-fähig. Unabhängig von der Generierungseigenschaft kann der Systembelegungs-modus vom Operator mit dem DISK-Parameter SYSTEM-ALLOCATION (siehe Kommando SET-DISK-PARAMETER) eingestellt werden. Die aktuelle Einstellung kann mit dem Kommando SHOW-DISK-STATUS (INF=*PAR/*ALL) abgefragt werden.

- SYS=*EXCL
  Die Platte wird system-exklusiv genutzt, andere Systeme werden von der Nutzung der Platte ausgeschlossen.
- SYS=*SHARE
  Die Platte wird system-shareable genutzt, bis zu drei weitere Systeme können die Platte gleichzeitig belegen.
- SYS=*ALL
  Der Systembelegungsmodus (system-exklusiv oder system-shareable) einer Privatplatte wird erst bei ihrer Belegung in Abhängigkeit von folgenden Faktoren bestimmt:
  - Art der Benutzeranforderung (task-shareable, task-exklusiv)
  - beabsichtigte Nutzungsart der Platte (DRV, SRV)

Zu einer system-exklusiven Belegung kommt es in jedem Fall dann, wenn die Platte von einem Benutzer task-exklusiv angefordert wird oder wenn es sich um eine DRV-Platte handelt.

Zu einer system-shareablen Belegung kommt es bei dieser Parametereinstellung in jedem Fall dann, wenn die Belegung von einem Benutzer-Job für task-shareable Benutzung bzw. vom Operator über das Kommando SET-DISK-PARAMETER ...,ASS=*OPER verlangt wurde und es sich um keine DRV-Platte handelt.



Bei Einsatz des Produkts MSCF und einer entsprechenden Hardware-Konfiguration ist der gleichzeitige und gemeinsame Zugriff über mehrere Systeme hinweg auf einen Satz **gemeinschaftlicher** Platten (Shared-Pubset) möglich.

Maximal 16 Systeme, die in einem gemeinsamen MSCF-Verbund gekoppelt werden, können über einen direkten Hardware-Pfad als „Sharer“ auf diesen mehrbenutzbaren Pubset zugreifen. Einer dieser Verbund-Teilnehmer wird zum temporären Eigentümer des Pubsets - zum „Pubset-Master“ - ernannt und wickelt für die anderen Systeme die Funktionen zur Verwaltung der Dateien, der Benutzer und der Zugriffe ab. Alle Verwaltungs-Anforderungen seitens der untergeordneten Teilnehmer, der sog. „Pubset-Slaves“ (auch: „Slave-Sharer“), müssen über MSCF an den Pubset-Master gerichtet werden.

Bei Ausfall des Pubset-Masters wird an allen abhängigen Systemen eine pubset-spezifische Jobvariable gesetzt. Die Systembetreuung kann daraufhin an diesen Systemen den Pubset exportieren und bei einem nachfolgenden IMPORT-PUBSET einen neuen Pubset-Master bestimmen.

Das gesamte Konzept der Shared-Pubsets (Hardware-Konfiguration, Verwaltung der Pubsets, Datenzugriffe) ist ausführlich im Handbuch „HIPLEX MSCF“ [33] beschrieben.

### 5.6.3 Weitere Vorgaben für die Privatplattenbelegung

**ASSIGN-TIME**

Mit dem DISK-Parameter ASSIGN-TIME kann der Operator Vorgaben einstellen für den Zeitpunkt der Belegung bzw. Freigabe einer Privatplatte sowie eines Plattengerätes mit entsprechender Aktualisierung des Systembelegungs-Protokolls auf der Platte.

- Mit ASS=*USER beginnt die Belegung der Platte und des Gerätes, auf dem sie montiert ist, mit der Belegung der Platte durch den Benutzer. Gleichzeitig wird die System-Id des Systems ins Systembelegungs-Protokoll eingetragen. Die Freigabe der Platte und des Gerätes erfolgt mit der Freigabe der Platte durch den Benutzer, das Systembelegungs-Protokoll wird bereinigt.

- Mit ASS=*OPERATOR erfolgt die Belegung der Platte sowie des Gerätes, auf dem sie montiert ist, unabhängig von der Belegung durch den Benutzer. Ist die Platte bereits online, so wird sie und das Gerät sofort belegt, sonst nach erfolgtem Einschalt-Interrupt, spätestens jedoch bei einer Belegungsanforderung durch einen Benutzer. Die Platte und das Gerät bleiben solange belegt, bis der Operator mit der Eingabe ASS=*USER die Belegung zurückgibt und alle Benutzer die Platte freigegeben haben.

  Platten, auf die zumindest über einen längeren Zeitraum von einem System zugegriffen werden soll, sollten über ASS=*OPERATOR belegt werden. Damit werden unnötige Belegungs- und Freigabeprotokolle vermieden.
  Bei der Belegung mit ASS=*OPERATOR ist zu beachten, dass vor dem Wegschalten mit dem Kommando DETACH-DEVICE ...,FORCE=*NO die Einstellung ASS=*USER vorzunehmen ist, da das DETACH-DEVICE-Kommando sonst abgewiesen wird. Nur so ist eine korrekte Bereinigung des Belegungsprotokolls sichergestellt.

*Belegungsbeginn*

- bei ASS=*USER mit der ersten Benutzeranforderung
- bei ASS=*OPERATOR
  - mit dem Einschalt-Interrupt bei Montierung (ATTACH-DEVICE)
  - sofort, wenn Platte bereits online
  - im Rahmen der ersten Belegungsanforderung durch den Benutzer

*Belegungsfreigabe*

- bei ASS=*USER mit der letzten Benutzerfreigabe
- bei ASS=*OPERATOR
  - bei bzw. nach SET-DISK-PARAMETER ASSIGN-TIME=*USER
  - bei bzw. nach SET-DISK-PARAMETER ASSIGN-TIME=*STD (und Standardwert ASSIGN-TIME=*USER eingestellt)
  - wenn keine Task die Platte mehr belegt

**OPERATOR-CONTROL**

Mit diesem DISK-Parameter legt der Operator fest, ob er Erst-Belegungen von Platten durch Tasks kontrollieren will. Schaltet der Operator die Kontrolle für eine Platte ein, so wird bei jeder ersten Belegung der Platte durch eine Task folgende Meldung ausgegeben:

```
NKA0004 ALLOCATION OF DISK '(&00)' IN USAGE MODE '(&01)' BY USER TASK
        PERMITTED? REPLY (Y=ALLOCATION ACCEPTED; N=ALLOCATION REJECTED)
```

Der Operator wird dadurch aufgefordert, zu entscheiden, ob die angegebene Platte der anfordernden Task in der Belegungsart (SHARED/EXCLUSIVE) zugewiesen werden soll. Nur wenn diese Meldung positiv (`TSN.Y`) beantwortet wird, wird die Platte von der Task belegt, sonst wird dieser und alle folgenden Belegungsversuche dieses Jobs ohne weitere Operatoraktion abgewiesen.

Ist eine Platte, für die der DISK-Parameter OPERATOR-CONTROL gesetzt wird, zu diesem Zeitpunkt bereits von einem Job belegt, so wird für diese Platte erst dann die Meldung `NKA0004` ausgegeben, wenn sie vom belegenden Job freigegeben wird und wieder neu belegt werden soll.

### 5.6.4 Steuerung der Betriebsmittelreservierung

Wird von einer Task ein Reservierungsauftrag gegeben (mit dem Kommando SECURE-RESOURCE-ALLOCATION), versucht das System, alle angeforderten Betriebsmittel (Geräte, Datenträger oder Dateien) für diese Task zu reservieren.
Sind die angeforderten Betriebsmittel verfügbar und können für die Task reserviert werden, so ist der Reservierungsauftrag abgeschlossen und die Bearbeitung weiterer Aufträge kann erfolgen.

Kann der Reservierungsauftrag nicht oder nur teilweise ausgeführt werden, wird der Job in eine Warteschlange, die Secure-Queue, eingetragen. Wartende Aufträge halten keine Betriebsmittel (Ausnahme ist ggf. die Collector-Task).

#### Die Collector-Task

Der Operator hat die Möglichkeit, jeweils einen Job in der Secure-Queue privilegiert zu behandeln. Diese Privilegierung besteht darin, dass diese Task am Anfang der Queue steht und die angeforderten Betriebsmittel sammeln kann. Diese Task wird als Collector-Task bezeichnet. Zur Steuerung der Collector-Auswahl stehen dem Operator die Kommandos START-RESOURCE-COLLECTION, MODIFY-RESOURCE-COLLECTION und STOP-RESOURCE-COLLECTION zur Verfügung.

Mit diesen Kommandos kann der Operator folgende Funktionen durchführen:

- Starten der COLLECTOR-Auswahl

  Mit dem Kommando START-RESOURCE-COLLECTION schaltet der Operator die Auswahl einer Collector-Task ein. Das System berechnet für jede Task in der Secure-Queue ein Gewicht nach der Formel:

  ```
  W = T + N * U
  ```

  `W` Weight; Gewicht

  `T` Time; Wartezeit, welche die Task bereits in der Secure-Queue verweilt

  `U` Urgency; dieser Wert berechnet sich aus der Task-Priorität, wobei die Task mit der höchsten Priorität die niedrigste Urgency hat und umgekehrt

  `N` Faktor, der vom Operator im Operanden TIME-WEIGHT des Kommandos START-RESOURCE-COLLECTION eingegeben wird (Standardwert ist 10).
  Durch gezielte Wahl des Faktors N kann der Operator die Berechnung der Gewichte beeinflussen:
  Wird N klein gewählt, bekommt die Wartezeit, die eine Task bereits in der Secure-Queue verweilt, große Bedeutung.
  Für N=0 ist die Wartezeit gleich dem Gewicht, die „Urgency“ hat keinen Einfluss auf die Berechnung.

Wird N jedoch groß gewählt, steigt die Bedeutung der Task-Priorität für die Berechnung des Gewichtes. Für N=600 berechnet sich das Gewicht fast nur aus der „Urgency“; die Wartezeit fällt bei der Berechnung kaum ins Gewicht.

Nachdem das System für alle Tasks in der Secure-Queue diese Berechnung durchgeführt hat, wird diejenige Task Collector-Task, die das höchste Gewicht hat. Die Collector-Task wird an den Anfang der Secure-Queue gestellt und kann ihre angeforderten Betriebsmittel sammeln. Wenn die Collector-Task alle von ihr angeforderten Betriebsmittel sammeln konnte, wird sie aus der Secure-Queue herausgenommen und das System berechnet für die in der Queue verbliebenen Tasks erneut die Gewichte, worauf eine neue Task zur Collector-Task wird.

- Tasks von der Auswahl ausschließen

  Der Operator kann mit dem Kommando MODIFY-RESOURCE-COLLECTION ACTION=*REMOVE-COLLECTOR eine Task von der Auswahl ausschließen. Wenn das System eine neue Collector-Task bestimmt (siehe Punkt 1), dann wird diese Task nicht berücksichtigt, d.h. sie kann nicht zur Collector-Task werden. Ist die angegebene Task zum Zeitpunkt des Ausschlusses Collector-Task, so verliert sie ihre Collector-Eigenschaft und damit alle bis dahin reservierten Betriebsmittel.

- Tasks zur Auswahl zulassen

  Der Operator kann eine vorher ausgeschlossene Task (siehe Punkt 2) wieder zur Auswahl zulassen (mit dem Operanden ACTION=*ADD-COLLECTOR); diese Task wird bei der nächsten Collector-Auswahl durch das System wieder berücksichtigt.

- Eine Task zur Collector-Task erklären

  Der Operator kann mit dem Kommando MODIFY-RESOURCE-COLLECTION ACTION=*SET-COLLECTOR eine Task zur Collector-Task machen. Ist eine andere Task Collector-Task, so verliert diese die Collector-Eigenschaft und damit die bereits gesammelten Betriebsmittel; die im Kommando angegebene Task wird zur Collector-Task. Diese Funktion ist auch ohne vorhergehendes START-RESOURCE-COLLECTION-Kommando möglich.

- Beenden der Collector-Auswahl

  Mit dem Kommando STOP-RESOURCE-COLLECTION beendet der Operator die mit dem Kommando START-RESOURCE-COLLECTION (siehe Punkt 1) gestartete Collector-Auswahl wieder. Ist eine Task Collector-Task, so verliert diese ihre Collector-Eigenschaft nicht, sie kann weiter die von ihr angeforderten Betriebsmittel sammeln. Entzieht der Operator dieser Task die Collector-Eigenschaft (siehe Punkt 2) oder kann die Collector-Task alle angeforderten Betriebsmittel sammeln, so wird vom System keine neue Collector-Task ausgewählt.

### 5.6.5 Hilfestellung im Umgang mit NDM

NDM bietet dem Operator ein breites Spektrum an Steuerungsmöglichkeiten für Datenträger. Da diese Parametereinstellungen unabhängig vom Online-Zustand (also auch vom Demontieren des entsprechenden Datenträgers) gelten, kann es beim Nichtbeachten der momentanen Einstellung der Parameter zu einem unerwarteten Systemverhalten kommen (z.B. kann eine Platte unter Umständen nicht belegt werden, obwohl sie schon montiert ist; soll die Belegung der Platte wieder zugelassen werden, muss der Operator den entsprechenden Parameterwert ändern). Alle Werte sind über Informationsfunktionen abfragbar.

Weitere unerwartete Reaktionen können eintreten, wenn der Zustand der Hardware nicht mit dem der Software übereinstimmt. (Wird z.B. ein Band auf ein Gerät im Zustand ATTACHED montiert, die Bandsteuerung ist jedoch nicht an den Server geschaltet, kann das Band nicht belegt werden.)

Im Folgenden werden die wichtigsten Punkte des Systemverhaltens beschrieben und es wird gezeigt, wie sich die entsprechende Situation in der SHOW-Ausgabe darstellt. Außerdem werden Hinweise gegeben, wie solche Situationen bewältigt werden können.

Folgende Situationen werden beschrieben:

- Mount-Meldung trotz montiertem Datenträger (Seite 227)
- SVL-Belegung durch andere Systeme (Seite 229)
- VSN-Mehrdeutigkeit für dieselbe Platte (Seite 230)
- Permanenter Hardwarefehler für belegte Geräte (Seite 231)
- Privatplatte permanent von einem anderen System gesperrt (Seite 233)
- Prüfung auf freie Platten (Seite 233)
- SPD-Handling (Seite 234)
- Ausschluss des SPD-Betriebes auf einem SPD-fähigen Gerät (Seite 235)
- SECURE-Deadlock-Situationen (Seite 235)
- Änderung der USER-ALLOCATION (Seite 237)
- Information über Reservierungen (Seite 237)
- Vorschläge zur performancegünstigen Einstellung von NDM-Parametern (Seite 238)

### Mount-Meldung trotz montiertem Datenträger

Es kann vorkommen, dass eine Mount-Anforderung an der Bedienstation ausgegeben wird, obwohl der Operator den angeforderten Datenträger bereits montiert hat. Dies kann folgende Ursachen haben:

1. Mehrere Datenträger gleicher VSN sind online (SHOW-DISK-STATUS, SHOW-TAPE-STATUS). Der Operator muss entscheiden, welcher Datenträger belegt werden soll.
2. Das Gerät, auf dem der Datenträger montiert wurde, ist von einer anderen Task belegt (SHOW-DEVICE-STATUS). Der Datenträger muss umgehängt werden.
3. Konfigurationszustand des Gerätes (es ist explizit oder implizit detached) verhindert die Belegung:
   a) Steuerung oder Gerät ist detached (SHOW-DEVICE-STATUS),
   b) Pfad ist (teilweise) removed (SHOW-DEVICE-CONFIGURATION).

   Datenträger umhängen oder Gerät/Pfad zuschalten (ATT/INC).
4. Der Hardwarezustand des Gerätes verhindert die Belegung:
   a) Gerät ist noch nicht eingeschaltet (Platte),
   b) Band ist noch nicht auf Bandanfangsmarke,
   c) Steuerung ist nicht zugeschaltet.
5. Für Server mit SVP-Bedienung:
   Pfad evtl. (von einem anderen Server aus) durch SVP-Kommando unterbrochen.
   Pfad über SVP wieder zuschalten oder Datenträger umhängen.
6. Einschalt-Interrupt für montierten Datenträger wurde von der Hardware nicht geliefert. Lesen der VSN kann durch CHECK-DISK-/CHECK-TAPE-MOUNT erzwungen werden.
7. Der montierte Datenträger besitzt eine andere VSN als die vom Benutzer spezifizierte (SHOW-DEVICE-STATUS, SHOW-DISK-STATUS, SHOW-TAPE-STATUS).
8. Weitere Ursachen bei Plattenanforderungen:
   a) Der vom Benutzer angegebene Gerätetyp stimmt nicht mit dem Typ der montierten Privatplatte überein (SHOW-DISK-STATUS VOL=vsn,INF=*PAR). Mount-Meldung abweisen bzw. ggf. zweite Platte gleicher VSN mit anderem Gerätetyp zuweisen.
   b) VOLIN: UNIT-Vorgabe verlangt ein anderes Gerät als das, worauf die Platte montiert wurde. Für eine Fortsetzung muss der Datenträger umgehängt werden oder die Mount-Meldung abgewiesen und der Benutzer zur Freigabe des Gerätes veranlasst werden.

c) Die anfordernde Task hat sich das Gerät, auf dem die Platte montiert wurde, mittels /SECURE-RESOURCE-ALLOCATION UNIT=mn belegt und will die Platte für DVS-Betrieb belegen (USE=*DMS). Da Platten mit UNIT-Anforderungen nur für Benutzungsart SPECIAL reserviert werden, muss die Mount-Meldung abgewiesen und der Benutzer zur Freigabe des Gerätes veranlasst werden.

d) Eine Platte kann auf Grund ihrer SVL-Belegung durch andere Systeme nicht belegt werden: Beschreibung der möglichen Reaktionen siehe Abschnitt „SVL-Belegung durch andere Systeme“ auf Seite 229).

e) Es soll eine Platte belegt werden, die während des letzten Systemlaufs als DRV-Platte eingesetzt war, und das Subsystem DRV ist (noch) nicht geladen.
Als Reaktion ist entweder das Subsystem DRV zu laden und die Mount-Meldung abzuweisen, um dem Benutzer einen neuerlichen Belegungsversuch zu ermöglichen, oder durch positive Beantwortung der Mount-Meldung einen SRV-Betrieb der Platte zu erzwingen.

9. Weitere Ursachen bei Bandanforderungen:

a) MODIFY-MOUNT-PAR ALLOC=*NO ist eingestellt (SHOW-MOUNT-PAR):
Die Mount-Meldung muss beantwortet werden.

b) Die vom Benutzer angegebene Schreibdichte wird von dem Gerät, auf dem der Datenträger montiert ist, nicht unterstützt (SHOW-RESOURCE-ALLOCATION, SHOW-DEVICE-CONFIGURATION). Band muss umgehängt werden.

c) Der Benutzer hat ein anderes Gerät explizit reserviert (SECURE-RESOURCE-ALLOCATION UNIT=mn) als das, worauf das Band montiert wurde.
Die SECURE-UNIT-Anforderung des Benutzers ist zwingend, d.h. das Band muss auf das angegebene Gerät montiert werden.

### SVL-Belegung durch andere Systeme

Obwohl eine Platte bereits als online protokolliert wurde, kann eine Belegungsanforderung durch einen Benutzer zu einer Mount-Meldung für dieselbe Platte führen. Dies ist dann der Fall, wenn eine Inkonsistenz zwischen der gültigen SYSTEM-ALLOCATION der Platte und ihrer tatsächlichen SVL-Belegung durch andere Systeme erkannt wird. Durch die Mount-Meldung, der ein Hinweis auf die vorliegende Diskrepanz vorausgeht, soll der Operator zu einer Entscheidung bzw. Reaktion aufgefordert werden.

Im Folgenden werden die möglichen Reaktionen und Eingriffe des Operators erläutert. Ausgangspunkt für die Überlegungen ist die Ausgabe von SHOW-DISK-STATUS VOL=vsn,INF=*ALL

1. Alle im SVL hinterlegten Systeme (SYSTEMS) sind nicht mehr aktiv:
   Die eingetragenen Systeme können mit UNLOCK-DISK VOL=vsn,SYS-ID=(...) ausgetragen werden. Die Mount-Meldung wird danach automatisch beantwortet, die Belegung wird dadurch akzeptiert.
2. Eingetragene Systeme arbeiten (teilweise) noch mit der Platte und die Platte soll vom eigenen System system-exklusiv belegt werden:
   Die Mount-Meldung muss abgewiesen werden.
3. Eingetragene Systeme arbeiten (teilweise) noch mit der Platte und die Platte soll vom eigenen System system-shareable belegt werden:
   a) SVL-ALLOC=EXCL:
      Die Mount-Meldung muss abgewiesen werden, da die Platte nicht gleichzeitig von einem weiteren System belegt werden kann.
   b) SVL-ALLOC=SHARE und alle Systeme aktiv:
      Im SVL sind 4 Fremdsysteme (bei Privatplatten) bzw. 16 Fremdsysteme (bei gemeinschaftlichen Platten) hinterlegt:
      Verfahrensweise in diesem Fall wie in Punkt 3a)
   c) SVL-ALLOC=SHARE und ein Teil der eingetragenen Systeme ist nicht mehr aktiv:
      Die inaktiven Systeme können – wie in Punkt 1 beschrieben – ausgetragen werden.
      Weitere Verfahrensweise wie in Punkt 3b).

### VSN-Mehrdeutigkeit für dieselbe Platte

Zur eindeutigen Identifizierung einer Platte verwendet die Plattenüberwachung neben der VSN auch den auf der Platte hinterlegten Timestamp (Datum und Uhrzeit, zu der die Platte vom zeitlich ersten im SVL eingetragenen System belegt wurde).

Eine Privatplatte auf einem SPD-fähigen Plattengerät wird vom System A system-exklusiv belegt (andere Systeme werden von der Benutzung der Platte ausgeschlossen). Eine Belegungsanforderung des Systems B für diese Platte wird somit abgewiesen. In der Annahme, System A arbeite nicht mehr mit der Privatplatte, trägt der Operator des Systems B mit dem Kommando UNLOCK-DISK VOL=PRIV01,SYS-ID=A die Belegung vom System A aus und belegt anschließend die Platte system-shareable (SPD-Betrieb möglich) vom System B aus. Durch diese „Erstbelegung“ der Platte wird auch der Timestamp neu geschrieben. Bei einer SVL-Überprüfung der Platte vom System A aus wird der veränderte Timestamp festgestellt.

Da die Platte vom System A aus mit dem „alten“ Timstamp als belegt gilt, wird dieselbe Platte jetzt als zweite Platte gleicher VSN, jedoch mit unterschiedlichem Timestamp betrachtet (VSN-Mehrdeutigkeit) und der Operator mit einer Remount-Meldung aufgefordert, die „alte“ Platte wieder einzuhängen. Die Platte kann auf Grund der Aktion von System B nicht mehr für System A zugewiesen werden.

Folgende Maßnahmen machen die Platte für System A wieder verfügbar:

1. Die Remount-Meldung muss mit `TSN.N` abgewiesen (CANCEL der Platte) und die belegenden Tasks zur Freigabe der Platte veranlasst werden (Abbau des CANCEL-Zustandes).
2. Nach Freigabe der Platte muss deren Belegung durch Benutzer wieder zugelassen werden (Kommando SET-DISK-PARAMETER UNIT=mn,USER=*ALL).
   Bei SPD-Geräten muss zusätzlich die Platte zum SPD-Betrieb zugelassen werden, um diese auch während eines Zugriffs von System B aus durch das System A belegen zu können (Kommando SET-DISK-PARAMETER VOL=vsn,SYS=*SHARE).
   Danach kann die Platte von System A aus wieder belegt werden.

### Permanenter Hardware-Fehler für belegte Geräte

Für den Fall, dass ein Band- oder Plattengerät, das durch einen montierten und von einem oder mehreren Benutzern verwendeten Datenträger belegt ist, laufend Hardware-Fehler (INOP,...) meldet, bieten sich folgende Reaktionsmöglichkeiten an:

1. Datenträger ist eine Festplatte:

   Kann der Hardware-Fehler nicht behoben werden, muss für eine Privatplatte die Remount-Meldung abgewiesen werden (impliziter Datenträger-Cancel); handelt es sich um eine gemeinschaftliche Platte, kommt der BS2000-Betrieb mit diesem Pubset zum Erliegen.

2. Datenträger ist ein Band:

   Der betroffene Datenträger soll auf einem anderen Gerät montiert werden (Kommando CHANGE-TAPE-MOUNT VOLUME=vsn,ACTION=*MOVE). Falls noch ein freies Gerät verfügbar ist, wird dem Operator mit der Remount-Meldung ein Ersatzgerät vorgeschlagen. Wird das Kommando abgewiesen, kann dies folgende Ursachen haben:

   – Der Belegung des Datenträgers ging ein SECURE-RESOURCE-ALLOCATION UNIT=mn voraus, dadurch bleibt der Datenträger während seiner Benutzung durch die belegende Task zwingend dem Gerät zugeordnet. Kann der Hardware-Fehler nicht behoben werden, muss die Remount-Meldung mit `TSN.N` beantwortet werden. Der Datenträger wird gecancelt.

   – Es ist kein Ersatzgerät mehr verfügbar.

   In jedem Fall muss das Gerät mit DETACH-DEVICE UNIT=mn,FORCE=*YES weggeschaltet werden.

   Das Wegschalten des Gerätes kann folgende Wirkungen erzielen:

   – Der Datenträger wird implizit gecancelt, sofern er ein Nicht-Standard-Label besitzt oder der Belegung ein SECURE UNIT=*mn vorausging.

   – Der Datenträger wird in den ACTION-Status „NO DEVICE“ überführt, d.h. sobald wieder ein freies Gerät existiert, wird dies dem privaten Datenträger zugeordnet.

   Beim Wegschalten von Geräten mit wechselbaren Datenträgern (Langbandgeräte, MBKs) wird in allen Phasen der Nutzung ein Warten auf das Freiwerden eines geeigneten Ersatzgerätes unterstützt, also sowohl im Zustand IN-USE als auch während der Bereitstellung (Zustand MOUNT oder PREMOUNT) sowie für reine Gerätetypbelegungen, z.B. nach SECURE-RESOURCE-ALLOCATION DEVICE=(TYPE=TAPE-C4, NUMBER=1).

Dabei kann der ACTION-Status „NO DEVICE“ für Datenträger in Benutzung oder während der Bereitstellung unmittelbar bei DETACH UNIT=mn,FORCE=*YES eintreten, wenn kein geeignetes Ersatzgerät verfügbar ist. Es kann ein „freies“ Gerät in dem Sinn verfügbar sein, dass diesem zwar momentan kein Datenträger zugeordnet ist, für das aber eine sog. Typbelegung existiert. Diese Belegung wird verdrängt.

Die betroffene Task wartet – während der Phase der Bereitstellung des mit dieser Typbelegung verknüpften Datenträgers – auf das Freiwerden eines geeigneten Ersatzgerätes.
Datenträger im Zustand IN-USE verdrängen Datenträger im Zustand PREMOUNT.
Das Verdrängen eines Datenträgers im Zustand MOUNT wird nur bei manuell bedienten Geräten automatisch durchgeführt. Bei von ROBAR unterstützten Geräten wird ein einmal eingeleiteter Montiervorgang nicht mehr abgebrochen, d.h. die Remount-Recover-Routine wird mit NO DEVICE terminiert und der Mount-Vorgang wird abgeschlossen (Eigenschaften von ROBAR).

Bei Freiwerden eines Gerätes durch ATTACH-DEVICE oder durch Belegungsende eines anderen Datenträgers erfolgt die Auswahl des zuzuordnenden Datenträgers in Abhängigkeit von der jeweiligen Phase (und zwar in folgender Reihenfolge: IN-USE vor MOUNT vor PREMOUNT), wenn mehr als ein Datenträger im Zustand NO DEVICE ist, der auf diesem Gerät vom Typ und der Lagerortzuordnung her verarbeitet werden könnte.

### Privatplatte permanent von einem anderen System gesperrt

Eine Platte kann nicht belegt werden, falls ein anderes System als VTOC-Lockhalter im SVL der Platte hinterlegt ist (siehe Feld „VTOC-SYS“ bei SHOW-DISK INF=*SYS).
Der Operator wird in folgenden Fällen über diesen Zustand informiert:

- Belegungsaufforderung durch den Benutzer führt zu Meldung `NDV0002` (Platte permanent durch ein anderes System gesperrt).
- Impliziter UNLOCK für die eigene System-Id bei Einhängen der Platte bzw. expliziter UNLOCK (UNLOCK-DISK VOL=vsn,SYS-ID=sys-id) führt ebenfalls zu Meldung `NDV0002`.
- SET-DISK-PARAMETER VOL=vsn,ASS=*OPER wird mit einem Hinweis auf den VTOC-Lockhalter abgewiesen.

Es muss sichergestellt werden, dass das als VTOC-Lockhalter eingetragene System nicht mehr mit der Platte arbeitet und der Lock auf Grund eines Fehlverhaltens (z.B. Systemabsturz) oder durch das Canceln der Platte zurückgeblieben ist. Nur für diesen Fall darf der Lockhalter ausgetragen werden; ansonsten muss auf die Freigabe des Locks durch das belegende System gewartet werden, da es sonst zu Nebeneffekten kommen kann (siehe Abschnitt „VSN-Mehrdeutigkeit für dieselbe Platte“ auf Seite 230).

Für die Punkte a) und b) kann ein Austragen des Lockhalters durch Beantwortung der Meldung `NDV0002` mit `tsn.F` erreicht werden.

Für Punkt c) muss das Austragen des belegenden Systems explizit mit UNLOCK-DISK VOL=vsn,SYS-ID=sys-id eingeleitet werden. Diese Anforderung führt ebenfalls wieder zur Meldung `NDV0002`, die wie zuvor beschrieben mit `tsn.F` beantwortet werden muss.

### Prüfung auf freie Platten

Ob eine Platte frei ist, wird z.B. dann geprüft werden, wenn sie neu initialisiert werden soll. Eine Platte wird nicht von Benutzern belegt, wenn das Kommando SHOW-DISK folgende Informationen liefert:

- PHASE=ONLINE
- PHASE=IN-USE und VOL-A=FREE bei Platten mit der Benutzungsart USE=DMS;
  In diesem Fall ist die Platte nur noch durch ihre ASSIGN-TIME belegt und kann durch SET-DISK-PARAMETER VOL=vsn,ASS=*USER[,USER=*NO] freigegeben werden.

### SPD-Handling

Im NDM sind Datenträger- und Geräteeigenschaft generell zu unterscheiden:

- die Eigenschaft „SPD-fähiges Gerät“ bezieht sich nur auf die Hardware-Eigenschaften eines Plattengerätes (Platten sind immer shareable generiert und damit SPD-fähig),
- die Eigenschaft „für SPD-Betrieb vorgesehen“ ist eine Einstellung (SET-DISK-PARAMETER SYS=*SHARE) für eine noch nicht belegte Privatplatte, die dem System vorgibt, in welchem Betriebsmodus die Platte zu belegen ist (andere Systeme sollen von der gleichzeitigen Nutzung der Platte nicht ausgeschlossen, Dateilocks sollen auf der Platte hinterlegt werden),
- die Eigenschaft „SPD-Platte“ bezeichnet eine schon belegte Privatplatte, die im SPD-Betrieb (Belegungsmodus system-shareable; SET-DISK-PARAMETER SYS=*SHARE) eingesetzt wird.

Eine Privatplatte kann auf einem SPD-fähigen Gerät nur dann als SPD-Platte belegt werden, wenn folgende Voraussetzungen gegeben sind:

1. Die Platte muss für den SPD-Betrieb vorgesehen sein (SET-DISK SYS=*SHARE oder SET-DISK SYS=*ALL für Platten auf einem Gerät mit der Eigenschaft POOL=SH montiert).
2. Die Platte darf von keinem anderen System im Modus system-exklusiv belegt sein (SVL-ALLOC muss ungleich „EXCL“ sein).
3. Maximal drei andere Systeme dürfen die Platte bereits belegen (ggf. verfahren, wie in Abschnitt „SVL-Belegung durch andere Systeme“ auf Seite 229 beschrieben).

*Kritische Punkte im Umgang mit SPD-Platten*

Die System-Id eines Systems, das von einem anderen Server aus noch mit der Platte arbeitet, darf nicht ausgetragen werden (siehe Abschnitt „VSN-Mehrdeutigkeit für dieselbe Platte“ auf Seite 230).

Eine SPD-Platte sollte immer durch die Einstellung ASSIGN-TIME=*OPERATOR belegt bleiben, da sonst bei Freigabe der Platte durch die Benutzer vom eigenen System auch die SVL-Belegung freigegeben wird und die Platte nach dem Belegen durch 4 Fremdsysteme nicht mehr vom eigenen System belegt werden kann.

### Ausschluss des SPD-Betriebes

Eine Privatplatte muss nicht zwingend im SPD-Modus betrieben werden. Standardmäßig wird eine Privatplatte bei der Belegung über das Kommando SET-DISK-PARAMETER mit ASS=*OPER oder bei der Belegung durch einen Benutzer-Job für eine task-shareable Benutzung zur SPD-Platte.

Soll dies vermieden werden, muss der Operator vor der Belegung der Platte mit SET-DISK-PARAMETER VOL=vsn,SYS=*EXCL eine explizite Einstellung der SYS-ALLOC veranlassen. Diese Einstellung bleibt so lange erhalten, bis sie wieder explizit verändert wird.

### SECURE-Deadlock-Situationen

Mit den Meldungen `NKS0022` und `NKS0054` wird der Operator auf Deadlock-Situationen hingewiesen, die bei der SECURE-Verarbeitung der aufgeführten Tasks festgestellt wurden. Diese zwei Meldungen weisen auf folgende Deadlock-Situationen hin:

1. `NKS0022` – Deadlock wegen Betriebsmittel, die die Tasks über die SECURE-Bearbeitung hinaus behalten dürfen:

   Eine Behebung der Deadlock-Situation ist nur über das Abbrechen einer oder mehrerer Tasks möglich.

   *Beispiel*

   TASK 1

   ```
   /CREATE-FILE FILE-NAME=DAT1,-
        SUPPORT=PRIVATE-DISK(VOLUME=-
        VOL1,DEVICE-TYPE=DEV1)

   /ASSIGN-SYSLST TO-FILE=DAT1
   :
   /SEC-RES DISK=(VOL=VOL2,-
        TYPE=DEV2,-
        ALLOC=EX),-
        WAIT=......
   ```

   TASK 2

   ```
   /CREATE=FILE FILE-NAME=DAT2,-
        SUPPORT=PRIVATE-DISK(VOLUME=-
        VOL2,DEVICE-TYPE=DEV2)

   /ASSIGN-SYSLST TO-FILE=DAT2
   :
   /SEC-RES DISK=(VOL=VOL1,-
        TYPE=DEV1,-
        ALLOC=EX),-
        WAIT=.......
   ```

   Sowohl Task 1 als auch Task 2 belegen für sich durch Öffnen einer Privatplattendatei (ASSIGN-SYSLST) jeweils eine Privatplatte task-shareable. Die SECURE-Anforderung der beiden Tasks zur task-exklusiven Belegung der von der jeweils anderen Task bereits task-shareable belegten Platte führt dazu, dass beide Tasks auf eine Plattenfreigabe warten.

   Lösung:<br>Diese klassische Deadlock-Situation kann nur durch das Beenden einer der beiden Tasks aufgehoben werden.

2. `NKS0054` – Deadlock wegen Collecting

Unter der Voraussetzung, dass während der laufenden Session eine Auswahl einer Collector-Task stattfindet, kann folgende Deadlock-Situation eintreten:

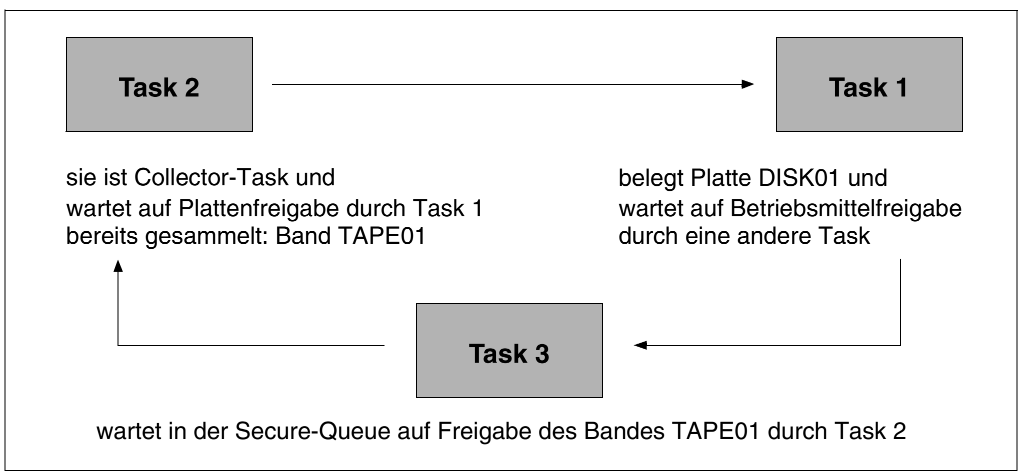



Bild 10: Deadlock-Situation wegen Collecting

Task 1 belegt die Platte DISK01 mit `/CREATE-FILE FILE-NAME=abc` und `/ASSIGN-SYSLST TO-FILE=abc` und wartet auf die Betriebsmittelfreigabe durch eine andere Task mit `/SECURE-RESOURCE-ALLOCATION FILE=(NAME=DATEI1,ALLOC=EX)`.

Task 2 ist die Collector-Task, sie wartet auf die Plattenfreigabe durch Task 1 mit `/SECURE-RESOURCE-ALLOCATION DISK=*PAR(VOL=DISK=01,TYPE=D3475,ALLOC=EX), TAPE=(VOL=TAPE01,TYPE=TAPE-C4),WAIT=...`

Task 3 wartet in der Secure-Queue auf die Freigabe des Bandes durch Task 2 während der Belegung der Datei DATEI1 `/SECURE-RESOURCE-ALLOCATION TAPE=(VOL=TAPE01,TYPE=TAPE-C4)`.

Lösung:
In diesem Fall würde eine Etablierung von Task 3 als Collector-Task zu einer Freigabe des Bandes TAPE01 durch Task 2 führen (Task 2 verliert seine Collector-Eigenschaft). Dadurch erhält Task 3 sein angefordertes Betriebsmittel und tritt aus der Secure-Queue aus. Nach Freigabe der Betriebsmittel durch Task 3 kann die Anforderung von Task 1 erfüllt werden; nach dessen Freigabe die von Task 2.

### Änderung der USER-ALLOCATION

Der Wert von USER-ALLOC für eine Platte entscheidet deren Belegbarkeit durch den Benutzer. Ist „NO“ eingestellt, werden alle Belegungsanforderungen ohne Hinweis für den Operator abgewiesen. Aus diesem Grund ist es für den Operator wichtig zu wissen, in welchen Fällen eine (implizite) Änderung des Parameters stattfindet.

1. SET-DISK-PARAMETER VOL=vsn,USER=....
2. Wenn für eine Platte USER-ALLOC=*STD definiert ist, gilt als tatsächlicher Wert derjenige, der mit SET-DISK-DEFAULTS USER=.... eingestellt wurde. Ändert sich also der Standardwert, ist auch diese Privatplatte davon betroffen.
3. Beim Entziehen einer Platte mit CHANGE-DISK-MOUNT VOL=vsn,ACTION=CANCEL wird USER-ALLOC=*NO eingestellt.
4. Das Beantworten einer Remount-Meldung mit `tsn.N` (Abweisung) hat einen impliziten Cancel der Platte zur Folge (siehe Punkt 3).

### Information über Reservierungen

1. Taskspezifisch

   Das Kommando SHOW-RESOURCE-ALLOCATION (SH-RES) gibt Auskunft über folgende Betriebsmittelbelegungen:

   – Unit-Reservierungen
   – Gerätetypreservierungen
   – Band-, Plattenreservierungen (explizit durch SECURE-RESOURCE-ALLOCATION oder implizit durch Eröffnen oder Reservieren von Dateien auf privaten Datenträgern)

2. Systemglobal

   Mit SHOW-DEVICE-STATUS UNIT=*SEL-TYPE(ATTR=*FREE[,TYPE=xxx]) erhält man eine Liste aller Geräte (eines bestimmten Typs), die noch nicht explizit (durch SECURE-RESOURCE-ALLOCATION UNIT=mn) oder implizit (durch Belegung eines darauf montierten Datenträgers) belegt wurden; diese Liste spiegelt aber nicht die tatsächliche Anzahl der noch belegbaren Geräte wider; es gibt nämlich Reservierungen, die noch keinem Gerät (Unit) zugeordnet sind (wie z.B. Typreservierungen mit dem Kommando SECURE-RESOURCE-ALLOCATION).

   Eine Übersicht über die tatsächliche Anzahl von reservierten Geräten und die Art ihrer Reservierung sowie die Anzahl der noch belegbaren Geräte eines Typs kann der Ausgabe des Kommandos SHOW-DEVICE-STATUS INF=*SUM[,TYPE=xxx] entnommen werden.

   SHOW-DEVICE-STATUS UNIT=*SEL-TYPE(TYPE=xxx), INF=*TASK zeigt die belegenden oder reservierenden Tasks und die Anzahl der von ihnen reservierten Geräte.

### Vorschläge zur perfomance-günstigen Einstellung von NDM-Parametern

1. Privatplatten, die im DVS-Betrieb eingesetzt werden, werden bei Dateibearbeitung auf Kommandoebene (COPY-FILE, DELETE-FILE, ADD-FILE-LINK, CREATE-FILE) durch das DVS immer wieder neu belegt und freigegeben. Haben diese Belegungen und Freigaben einer Task außerdem immer eine SVL-Neubelegung bzw. eine Freigabe der Privatplatte zur Folge, so muss neben der DVS-Ein-/Ausgabe zusätzlich immer noch eine SVL-Verwaltung (Ein-/Austragen der System-ID im SVL der Platte) stattfinden. Bleibt die Platte jedoch über die Freigabe durch den jeweiligen Benutzer hinaus noch belegt (durch weitere gültige Belegung eines anderen Benutzers oder durch ASSIGN-TIME=*OPERATOR), entfällt der nicht unerhebliche Aufwand für die SVL-Verwaltung. Aus diesem Grund sollte eine für den DVS-Betrieb eingesetzte Privatplatte so lange durch ihre ASSIGN-TIME belegt bleiben, bis sie nicht mehr gebraucht und gezielt vom Operator freigegeben wird. Erreichbar ist dies durch

   a) SET-DISK-PAR VOL=vsn,ASS=*OPER

   b) SET-DISK-DEFAULTS ASS=*OPER (gilt für alle Platten mit ASS=*STD)

   i Sollen auch SPECIAL-Anwendungen stattfinden, ist Alternative b) nicht zu empfehlen, da dadurch auch Platten, die für eine spätere SPECIAL-Nutzung montiert wurden, automatisch belegt werden und vor ihrer Bearbeitung durch die SPECIAL-Anwendung vom Operator explizit freigegeben werden müssen (mit SET-DISK-PAR VOL=vsn,ASS=*USER).

2. Für eine Privatplatte sollte nur dann SYS=*SHARE eingestellt werden, wenn auch ein echter Parallelbetrieb von mehreren Systemen darauf stattfinden soll (siehe auch Abschnitt „SPD-Handling“ auf Seite 234). In diesem Modus ist nämlich eine aufwändigere Dateilockverwaltung des DVS notwendig (Locks werden auf der Platte hinterlegt). Da Privatplatten immer shareable generiert sind und damit auch immer SPD-fähig sind, sollte für alle Platten, die nicht im SPD-Betrieb genutzt werden sollen, vor ihrer Belegung folgendes Kommando eingegeben werden: /SET-DISK-PARAMETER VOL=vsn, SYS=*EXCL

3. Für ein Data Center, bei dem der SPOOLOUT auf Band ausgelagert werden soll, sollte MODIFY-MOUNT-PAR UNLOAD=*NO (Voreinstellung) gesetzt sein, da ansonsten nach jedem Auslagern einer SPOOL-Datei das entsprechende Band entladen wird und ggf. vom Operator wieder startklar gemacht werden muss.

## 5.7 Datenträger-Überwachung

Die Geräteverwaltung unterstützt und überwacht die Platten- und Bandbenutzung, sie wickelt Montier- und Ummontiervorgänge ab und schützt benützte Bänder und Platten vor fehlerhaften Eingriffen.

Sie verwendet dazu zwei permanente Tasks, den DISK-MONITOR (DM) für die Plattenüberwachung und den TAPE-MONITOR (TM) für die Bandüberwachung. Für Funktionen, die die Verfügbarkeit der permanenten Tasks zeitweise beeinträchtigen (z.B. Bandpositionierungen), werden temporäre Tasks gestartet.

### Montieren eines Datenträgers vor der Benutzung bzw. bei Belegungsanforderung

Fordert ein Benutzer einen Datenträger an, so wird an der Bedienstation eine Mount-Meldung ausgegeben, falls dieser noch nicht als montiert erkannt wird (`NKVD013/NKVT013`; abhängig davon, ob eine Platte (D) oder ein Band (T) montiert werden soll).

Erkennt die Datenträger-Überwachung den angeforderten Datenträger als montiert (durch Auswertung des Einschalt-Interrupts vom Gerät), so wird die ausstehende Meldung vom System automatisch beantwortet. Für den Operator entfällt dadurch die Notwendigkeit, die Mount-Meldung zu beantworten.

### Ummontieren von Datenträgern während der Benutzung

Wird durch Rekonfiguration ein (defektes) Gerät weggeschaltet (Kommando DETACH), so wird der Operator bei Verfügbarkeit eines Ersatzgerätes von der Datenträger-Überwachung zu einem Ummontieren aufgefordert.

Stehen keine Ersatzgeräte zur Verfügung, so wird eine temporäre Überbelegung akzeptiert, d.h. es wird solange mit einer erneuten Mount-Aufforderung gewartet, bis ein Gerät frei oder zugeschaltet (Kommando ATTACH) wird. Der Operator kann jedoch dieses Warten durch expliziten Entzug des Datenträgers abbrechen (CANCEL-Funktion des Kommandos CHANGE-TAPE-MOUNT).

### Schutz von Datenträgern gegen fehlerhafte Geräteeingriffe und Repositionierung von Bändern

Jeder Eingriff des Operators an einem Band- oder Plattengerät (Entladen des Bandes, Ausschalten des Gerätes) mit einem benutzten Datenträger führt zu einer zeitweiligen Ein-/Ausgabe-Sperre für den Benutzer.

Wird bei einer Ein-/Ausgabe-Anforderung festgestellt, dass der Datenträger nicht mehr verfügbar ist, führt dies zu einer Aufforderung an den Operator, den Datenträger wieder bereitzustellen (Remount-Meldung).

Nach einer erneuten Zuweisung des Datenträgers können die Ein-/Ausgaben fortgesetzt werden. Bei Bändern findet eine automatische Repositionierung statt.
Werden den Monitoren Hardware-Resets der Geräte gemeldet, führt dies ebenfalls zur Identifizierung der Datenträger und bei Bändern mit PHASE=IN-USE zu deren Repositionierung.

**(Automatischer) Abbau einer Datenträgerüberbelegung**

Wird einem belegten Band im USE-MODE=DMS das Gerät entzogen (DETACH UNIT=..., FORCE=*Y), ohne ein Ersatzgerät zur Verfügung zu haben, entsteht eine temporäre Überbelegung (SH-TAPE: Ausgabespalte ACTION=NO DEVICE), d.h. es sind mehr Datenträger dieses Gerätetyps in Benutzung, als Geräte verfügbar sind (SHOW-TAPE-STATUS: Spalte PHASE=IN-USE).

Welcher Datenträger zeitweilig nicht verfügbar sein soll, kann der Operator ggf. durch das Kommando CHANGE-TAPE-MOUNT ...,EXCHANGE=(...) beeinflussen bzw. bestimmen.
Wird ein Gerät eines passenden Typs frei, so wird der Operator von der Datenträger-Überwachung automatisch zum Montieren des Datenträgers aufgefordert (mit ACTION=NO DEVICE).

## 5.7.1 Geräteauswahlmechanismus für Bandgeräte

### Allgemeine Geräteauswahl

Für die Auswahl durchsucht NDM die Gerätetabelle von Beginn an, d.h. in der bei der Hardware-Generierung festgelegten Reihenfolge, und wählt das erste passende, freie Gerät aus (BEST-GENERATED-DEVICE). Das hat zur Folge, dass die ersten in der Tabelle stehenden Geräte häufiger für eine Nutzung ausgewählt werden und – gerade in Verbindung mit roboterbedienten Magnetbandarchivsystemen wie ROBAR, die sich immer an den Mount-Vorschlag von NDM halten – eher abgenutzt werden, als Geräte am Ende der Tabelle.

Für Benutzer, die an einer gleichmäßigen Nutzung ihrer Bandgeräte interessiert sind, wird die Funktion „wrap around“ angeboten. Sie kann über das Kommando MODIFY-MOUNT-PARAMETER NEXT-TAPE-MOUNT=LEAST-RECENTLY-USED-DEVICE eingeschaltet werden. NDM wählt dann aus den passenden freien Geräten in der Gerätetabelle dasjenige aus, welches am längsten unbenutzt war.

Für Benutzer, die an einer auf den Ein-/-Ausgabe-Durchsatz hin optimierten Geräteauswahl interessiert sind, wird das Kommando MODIFY-MOUNT-PARAMETER ...,NEXT-TAPE-MOUNT=*BY-CONTROLLER angeboten. Aus den passenden freien Geräten kann ein Gerät ausgewählt werden, an dessen Steuerung die wenigsten Geräte in Benutzung sind. Damit wird eine gleichmäßige Verteilung der benutzten Geräte an den vorhandenen Steuerungen und Kanalpfaden erzielt. Die Geräteauswahl im NDM berücksichtigt nur die eigenen Gerätebelegungen. In Verbindung mit der IORM-Funktion DDAL (siehe Seite 254) wird die Optimierung auf alle VM2000-Gastsysteme eines Servers erweitert.

Die Standardeinstellung des Geräteauswahlmechanismus ist NEXT-TAPE-MOUNT= *BEST-GENERATED-DEVICE, d.h. NDM durchsucht die Gerätetabelle vom Beginn an, in der bei der Hardware-Generierung vorgegebenen Reihenfolge, und wählt das erste passende, freie Gerät aus.

### Geräteauswahl unter Berücksichtigung von Lagerorten

Insbesondere zur Unterstützung robotergesteuerter Archive in BS2000/OSD (Software-Produkt ROBAR, siehe Abschnitt „Archivsysteme“ auf Seite 559) wurde im NDM eine Lagerortverwaltung entwickelt.
Datenträger und die Geräte, auf denen sie montiert werden sollen, können in BS2000/OSD Lagerorten zugeordnet werden. Die Vereinbarung von Lagerorten ermöglicht es, Datenträger und Geräte aus einer genau vorgegebenen Menge auszuwählen und zu reservieren.

Die Zuordnung von Bandgeräten zu den Lagerorten erfolgt mit dem Kommando ADD-DEVICE-DEPOT. Es sind 1024 Bandgeräte pro Lagerort erlaubt. Die Zuordnung einer VSN zu einem Lagerort wird durch das Datenträger-Verwaltungssystem MAREN verwaltet.
Die dort festgelegten Lagerorte müssen mit den Angaben des Operators im Kommando ADD-DEVICE-DEPOT übereinstimmen.

Jeder Lagerort ist im MAREN-Katalog durch seinen Namen (max. 8 Zeichen), seinen Typ (REMOTE oder LOCAL) und seinen Bedienmodus gekennzeichnet.



Die Arbeit mit Lagerorten ist nur in Verbindung mit MAREN (siehe Seite 249) möglich.

Zum Verständnis der Auswirkungen, die das Kommando ADD-DEVICE-DEPOT auf die Geräteauswahl im NDM bei Belegungsanforderungen hat, werden im Folgenden die Grundprinzipien kurz beschrieben. Dabei wird nach der Art der Belegungsanforderung (Bandgeräte-Anforderung und Band-Volume-Anforderung) unterschieden.

### Geräteauswahl bei Bandgeräte-Anforderung

Die Anforderung eines Bandgerätes kann von Seiten des Benutzers auf drei Arten erfolgen:

1. Anforderung über den mnemotechnischen Gerätenamen (MN)

   Die Anforderung des Gerätes ist bei Bezeichnung eines mnemotechnischen Gerätenamens explizit vorgegeben; es erfolgt keine Auswahl.

2. Anforderung mit Angabe eines Lagerortes

   Die Bandgeräteanforderung erfolgt über die Angabe eines Geräte- oder Volumetyps mit zusätzlicher Angabe eines Lagerortes.

   Erfolgt die Anforderung über das Kommando SECURE-RESOURCE-ALLOCATION, berücksichtigt NDM bei der Auswahl nur den angegebenen Lagerort. Ist hier kein Gerät verfügbar, wird der Job des Benutzers abgewiesen.

   Erfolgt die Anforderung über eine interne Schnittstelle, die von ARCHIVE genutzt wird, musste der Aufrufer den Lagerort zuvor mit Hilfe von MAREN bestimmen. Falls dieser Lagerort dem NDM bekannt ist, d.h. über das Kommando ADD-DEVICE-DEPOT eingerichtet wurde, wählt NDM ein Bandgerät aus diesem Lagerort aus. Sonst erfolgt ein Zugriff auf die Gerätemenge, die den angegebenen Typ unterstützt, jedoch keinem bestimmten Lagerort zugeordnet ist. Diese Menge wird als „Restpool“ bezeichnet.

3. Anforderung ohne Angabe eines Lagerortes

   Die Bandgeräteanforderung erfolgt über die Angabe eines Geräte- oder Volume-Typs ohne zusätzliche Angabe eines Lagerortes.

   Erfolgt die Anforderung über das Kommando SECURE-RESOURCE-ALLOCATION z.B. mit dem Operanden TYPE=TAPE-C4,...,LOCATION=*USER-DEF, versucht NDM, über MAREN einen Standard-Lagerort für den angegebenen Volume-Typ oder einen von dem angegebenen Gerätetyp unterstützten Volume-Typ zu ermitteln.
   Kennt das NDM diesen Lagerort, so wählt es ein Bandgerät aus diesem Lagerort aus. Sonst erfolgt ein Zugriff auf den „Restpool“.

Erfolgt die Anforderung über das Kommando SECURE-RESOURCE-ALLOCATION mit dem Operanden LOCATION=*NONE, entfällt der MAREN-Aufruf und es wird unmittelbar auf den „Restpool“ zugegriffen.

**Implizite Geräteauswahl bei Band-Volume-Anforderung**

Die Anforderung eines Band-Volumes kann von Seiten des Benutzers auf zwei Arten erfolgen:

1. Anforderung mit Angabe der VSN

   Erfolgt die Anforderung mit Angabe der VSN, wird der in MAREN für diese VSN definierte Lagerort verwendet. Falls in MAREN für diese VSN kein Eintrag existiert, liefert MAREN entweder einen über die MAREN-Exit-Routine ermittelten oder einen Standard-Lagerort.

2. Anforderung ohne Angabe der VSN

   Die Band-Volume-Anforderung erfolgt ohne Angabe einer VSN.

   NDM versucht in diesem Fall, über MAREN einen Standard-Lagerort für den angegebenen Volume-Typ oder einen von dem angegebenen Gerätetyp unterstützten Volume-Typ zu ermitteln. Kennt das NDM diesen Lagerort, so wählt es ein Bandgerät aus diesem Lagerort aus. Sonst erfolgt ein Zugriff auf den „Restpool“.

## 5.8 PAV: Parallel Access Volume (S-Server)

Im BS2000/OSD wird jede Platte durch einen Tabelleneintrag in der PDT (Physical Device Table) repräsentiert. Jeder I/O-Auftrag für eine Platte wird vom Gerätetreiber in Form eines Kanalprogrammes an die I/O-Steuerung IOCNTRL zur Ausführung weitergereicht.

Wenn im PDT-Eintrag angezeigt wird, dass das Gerät nicht aktiv ist, wird der I/O-Auftrag gestartet. Jeder weitere I/O-Auftrag für dieses Gerät wird von IOCNTRL zunächst nur in eine Gerätewarteschlange eingereiht. Erst wenn das Gerät nicht mehr aktiv ist, wird der nächste Auftrag aus der Warteschlange gestartet.
Da zu einem Zeitpunkt also nur ein I/O-Auftrag pro Platte erlaubt ist, setzt sich die Gesamtdauer einer I/O zusammen aus der eigentlichen I/O-Dauer in der Hardware und der Wartezeit.

Hohe Wartezeiten können dadurch verursacht werden, dass:

- mehrere Anwendungen gleichzeitig mit hoher I/O-Last auf einer Platte arbeiten (siehe Bild unten).
- eine Anwendung ihre I/O-Aufträge asynchron an IOCNTRL abgibt.

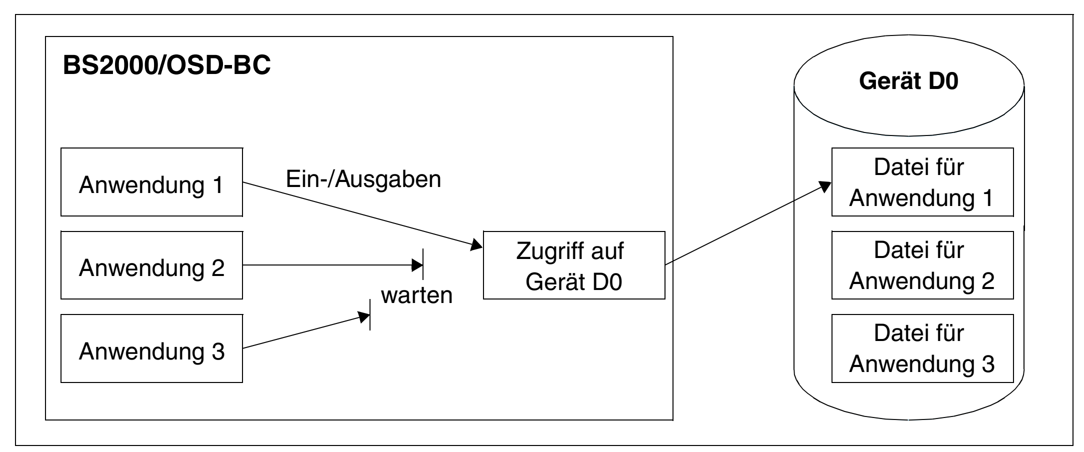



Bild 11: Mehrere Anwendungen mit I/O-Last auf einer Platte (traditionell)

Um lange Wartezeiten zu vermeiden, musste bisher der Datenbestand reorganisiert, d.h. häufig benötigte Daten von einer auf mehrere Platten verteilt werden.

> **i** Bei SQ-Servern kann IOCNTRL standardmäßig mehrere Aufträge für emulierte Platten parallel starten (RSC-Funktion). Es müssen keine besonderen Einstellungen getroffen werden.

### Symmetrix-Steuerungen am S-Kanal mit PAV

Die Lösung besteht darin, mehrere I/O-Aufträge gleichzeitig auf eine Platte auszuführen. Dazu bieten Symmetrix-Steuerungen die Funktion „Parallel Access Volume“ (PAV) an. Die Steuerung erlaubt mehrere gleichzeitige Zugriffe auf ein logisches Volume (CKD-formatiertes oder – ab Enginuity-Stand 5568 – auch FBA-formatiertes Plattengerät).

Die Parallel-Access-Volumes werden vom Systemservice in der Symmetrix installiert. Ein logisches PAV-Volume wird durch ein sog. Basis-Gerät (BASE) und bis zu 7 Alias-Geräten (ALIAS) repräsentiert.
Basis- und Alias-Geräte werden bei der Hardware-Generierung als eigenständige Geräte gleichen Typs mit unterschiedlichen Unit-Adressen und unterschiedlichen mnemotechnischen Gerätenamen, jedoch in derselben logischen Steuerung generiert.

Den Zusammenhang zwischen Basis-Gerät und Alias-Geräten ermittelt BS2000/OSD dynamisch beim Zuschalten (Kommando ATTACH-DEVICE) eines Gerätes. Erkennt BS2000/OSD ein Basis-Gerät, wird nach zugehörigen Alias-Geräten an derselben logischen Steuerung gesucht und diese implizit zugeschaltet. In diesem Zusammenhang werden die Alias-Geräte für eine nachfolgende Nutzung implizit „ready“ geschaltet.
Alias-Geräte können auch explizit mit ATTACH-DEVICE zugeschaltet werden.
Das Zuschalten eines Alias-Gerätes wird jedoch von BS2000/OSD verweigert, wenn das zugehörige Basis-Gerät nicht zugeschaltet ist.
Beim Wegschalten (Kommando DETACH-DEVICE) eines Basis-Gerätes werden zugehörige Alias-Geräte von BS2000/OSD implizit „not ready“ und weggeschaltet.
Ein Alias-Gerät kann auch explizit mit DETACH-DEVICE weggeschaltet werden.

Sobald ein Basis-Gerät zugeschaltet und die zugehörigen Alias-Geräte zugeschaltet und „ready“ sind, kann IOCNTRL I/O-Aufträge für das PAV-Volume parallel über Basis- und Alias-Geräte starten (siehe folgendes Bild).

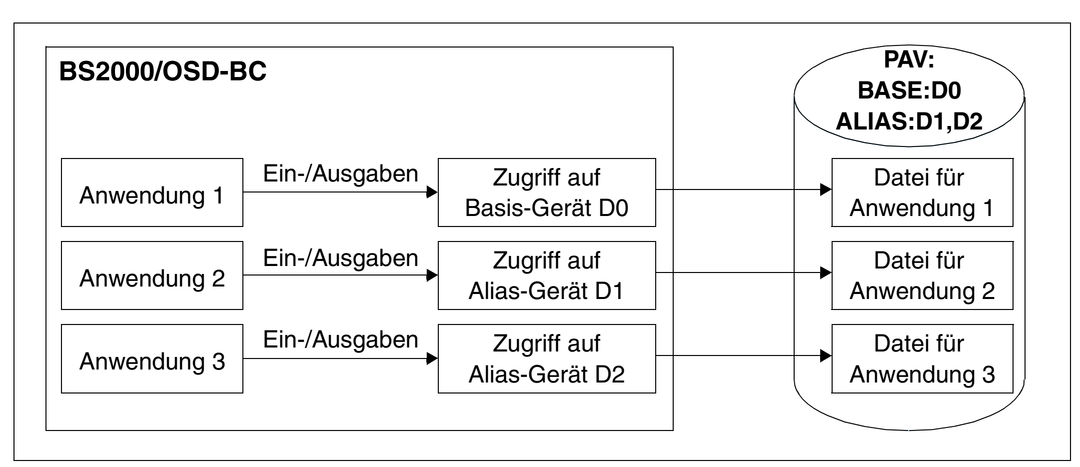



Bild 12: Mehrere Anwendungen mit I/O-Last auf einer Platte (mit Parallel Access Volumes)

Die Belegung eines PAV-Volumes (Kommando IMPORT-PUBSET bzw. SET-DISK-PARAMETER) erfolgt über das Basis-Gerät. Auch die Aufträge für Lese- oder Schreib-I/Os in BS2000/OSD erfolgen über das Basis-Gerät.
Erst IOCNTRL verteilt die I/Os in Abhängigkeit der Auslastung von Basis- und Alias-Geräten auf eines von diesen. Dabei bevorzugt IOCNTRL im native mode bei gleicher Auslastung das Basis-Gerät.

Vor Änderung einer Basis-/Alias-Beziehung in der Symmetrix muss das Basis-Gerät (und damit implizit alle Alias-Geräte) weggeschaltet werden. Nach der Änderung genügt ein Zuschalten des Basis-Gerätes für eine automatische Erkennung des neuen PAV-Volumes durch BS2000/OSD.

Parallel von BS2000/OSD gestartete Lese-I/Os auf ein PAV-Volume werden von der Symmetrix parallel ausgeführt, Schreib-I/Os nur bei disjunkten Extentangaben im Kanalprogramm.

Mit dem Kommando SHOW-DEVICE-CONFIGURATION ...,INFORMATION=*PAV können Informationen über alle PAV-Geräte ausgegeben werden.

*Hinweise zum Einsatz unter VM2000*

Um unter VM2000 eine möglichst gleichmässige Verteilung der I/Os für Shared-Platten zu erreichen, bevorzugt IOCNTRL in den Gastsystemen das Basis- oder ein Alias-Gerät in Abhängigkeit vom VM-Index der virtuellen Maschine.
Basis- oder Alias-Gerät können jedoch auch Gastsystem-spezifisch explizit definiert werden. Dazu steht das Kommando MODIFY-IO-UNIT UNIT-DEVICE=*DEVICE(NAME=..., STATE=*PAV-PREFERRED-DEVICE(...)) zur Verfügung. Für jedes Gastsystem kann ein bevorzugtes Alias-Gerät für die I/O-Ausführung festgelegt werden.

Vor Änderung einer Basis-/Alias-Beziehung in der Symmetrix (Plattengeräte am Typ-S-Kanal) muss das Basis-Gerät in allen Systemen, in denen es zugeschaltet ist, weggeschaltet (und der VM entzogen) werden.

Ausführliche Informationen zu PAV unter VM2000 finden Sie im Handbuch „VM2000“ [62].

### PAV am FC-Kanal

Für die Realisierung der PAV-Funktionalität nutzt BS2000/OSD die Tatsache aus, dass am FC-Kanal ein Gerät einen I/O-Auftrag bereits annehmen kann, während noch eine I/O aktiv ist. Im RAID-System darf jedoch das „Command Queuing“ nicht ausgeschaltet sein.

Im Gegensatz zu PAV für Symmetrix am S-Kanal, für die durch den Service an der Symmetrix-Steuerung die entsprechenden Einstellungen (Base/Alias-Devices usw.) vorgenommen werden müssen, ist PAV am FC-Kanal eine reine Software-Lösung, die ohne Eingriffe am RAID-System in Betrieb genommen werden kann. Der Einsatz von PAV am FC-Kanal der S-Server wird dringend empfohlen.

PAV am FC-Kanal kann für Plattenspeichersysteme ETERNUS DX400/8000 und Symmetrix verwendet werden.

Die Generierung von Alias-Geräten erfolgt mit IOGEN (siehe Handbuch „Systeminstallation“ [57]) oder im laufenden Betrieb mit dem Kommando ADD-IO-UNIT, Operand PAV-ALIAS-DEVICE.

### Einsatzempfehlungen für PAV

PAV bringt enorme Verbesserungen bei TP- und Batch-Betrieb, wenn ein großer Teil der Daten im Cache des Plattenspeichersystems liegt. Cache-Hits werden gleichzeitig bedient, parallel dazu kann ein Cache-Miss mit physikalischem Plattenzugriff ausgeführt werden. Für eine stark ausgelastete Platte mit n-pfadigem Anschluss kann es daher sinnvoll sein, n-1 Alias-Geräte zu generieren. Damit kann der n-fache Durchsatz einer Platte ohne Alias-Geräte erreicht werden.

Bei der Datensicherung muss davon ausgegangen werden, dass der Cache nicht ausreicht, die Daten also beim Sichern von den Platten geholt, bzw. beim Restore auf die Platten geschrieben werden müssen. Dies geschieht innerhalb des Plattenspeichersystems hoch-optimiert durch asynchrones „Read Ahead“ bzw. „Delayed Fast Write“. Die Leistung dieser Funktionen beschränkt den Durchsatz bei der Datensicherung.

ARCHIVE verwendet mehrere parallele asynchrone I/Os bei der Sicherung einer (großen) Datei. Mit PAV ergeben sich deutliche Verbesserungen am S-Kanal sowie bei schneller Plattenperipherie und Raid-Systemen mit vielen Platten, z.B. Raid10 (3+3) oder Raid5 (4+1). Mit langsamen Platten oder Raid-Systemen mit wenig Platten, z.B. Raid1 (1+1), werden trotz PAV keine Verbesserungen erreicht.

Trotz PAV reicht für Platten am S-Kanal der Durchsatz nicht aus, um schnelle Bandgeräte, z.B. LTO-Geräte, zum „Streamen“ zu bringen.

Für eine optimale Datensicherung wird daher ein Einsatz „großer“ Raid-Systeme mit schnellen Platten empfohlen sowie die Generierung von ausreichend vielen Alias-Geräten am S- oder FC-Kanal. Siehe auch die IORM-Funktion TCOM, Seite 254.

### Dynamisches PAV am FC-Kanal

Statisches PAV, wie zuvor beschrieben, verlangt eine vorausschauende Planung auf die zukünftige Geräteauslastung. In der Steuerung (Einrichten von Parallel Access Volumes am S-Kanal) und in BS2000/OSD (Generierung von Alias-Geräten am S- und FC-Kanal) müssen Vorkehrungen getroffen werden. Es ist natürlich auch möglich, allen Platten im Voraus ein oder mehrere Alias-Geräte zuzuweisen. Bei der Generierung im BS2000/OSD muss bedacht werden, dass für jedes Alias-Gerät eine eigene Adresse benötigt wird.
Wenn für jedes Gerät ein Alias-Gerät definiert wird, können für eine logische Steuerung nur noch 128 Geräte definiert werden, da maximal 256 Geräte an einer logischen Steuerung angeschlossen sein können.

Dynamisches PAV (die IORM-Funktion DPAV, siehe Seite 254) kommt mit weniger Alias-Geräten aus. Wie beim statischen PAV müssen Alias-Geräte generiert werden, es ist jedoch nicht erforderlich, für jedes Volume im Voraus die maximal benötigten Alias-Geräte vorzusehen. DPAV weist autonom Alias-Geräte denjenigen Volumes zu, die am meisten davon profitieren.

Dynamisches PAV unterstützt Extended PAV.

### Extended PAV (XPAV) am FC-Kanal

Mit der Funktion „Extended PAV (XPAV)“ für Geräte am FC-Kanal wird die bisherige Einschränkung, dass ein Alias-Gerät in derselben logischen Steuerung wie das Basis-Gerät liegen muss, aufgehoben. Zusätzlich zu einer vorhandenen logischen Steuerung können eine oder mehrere logische Steuerungen mit 256 Alias-Geräten konfiguriert werden.

XPAV eignet sich insbesondere für Konfigurationen, in denen PAV nicht von Anfang an eingeplant wurde oder in denen keine oder nur wenige freie Adressen für Alias-Geräte zur Verfügung stehen.

Zwei XPAV-Varianten sind möglich:

1. Wenn die vorhandene logische Steuerung und die zusätzliche Steuerung mit den Alias-Geräten an denselben Steuerungsports generiert werden, dann ist kein manueller Eingriff in der Plattensteuerung und auch keine zusätzliche Verkabelung erforderlich.

2. Wenn die vorhandene logische Steuerung und die zusätzliche Steuerung mit den Alias-Geräten an unterschiedlichen Steuerungsports generiert sind, dann müssen die Volumes in der Platten-Steuerung zusätzlichen Ports zugewiesen werden. Auch kann eine zusätzliche Verkabelung für die Alias-Geräte erforderlich werden.

Die Generierungs-Einschränkungen sind im Handbuch „Systeminstallation“ [57] beschrieben.

# 5.9 Verwaltung privater Datenträger

## 5.9.1 Einsatz von MAREN

Für die Verwaltung von privaten Datenträgern steht der Systembetreuung das Software-Produkt MAREN zur Verfügung. Dieses Software-Produkt dient zur Verwaltung von Datenträgerbeständen im Data Center. MAREN speichert alle Informationen über die Datenträger in einer ISAM-Datei, dem MAREN-Katalog.
Dieser Katalog kann von der Systembetreuung zentral für mehrere Server installiert werden, der Server-übergreifende Datentransfer wird im RFA- oder SPD-Modus realisiert.

Ohne MAREN ist das Arbeiten mit Lagerorten (siehe Seite 241) nicht möglich.

### MAREN-Katalog

Um die Informationen im MAREN-Katalog stets aktuell zu halten, ist das MAREN-System mit BS2000/OSD gekoppelt. Über diese Kopplung wird jeder DVS-Zugriff registriert und der MAREN-Katalog bei jeder Verarbeitung eines Magnetbandes aktualisiert. Zuvor werden jeweils diverse Prüfungen auf Zugriffsberechtigung, Verfügbarkeit und Plausibilität durchgeführt.

Der MAREN-Katalog ist VSN-orientiert. Die Archivnummern, auch unterschiedlicher Datenträgertypen, müssen eindeutig sein, wobei ein spezielles Nummernkreisschema von MAREN nicht vorausgesetzt wird.

Der MAREN-Katalog ist aus folgenden Gründen weder eine Kopie, noch Über- oder Untermenge des Dateikatalogs TSOSCAT:

- Der TSOSCAT ist Server- bzw. Pubset-bezogen, dagegen enthält der MAREN-Katalog Server-übergreifende Informationen.
- Falls im BS2000-Dateikatalog die Katalogeinträge privater Datenträger gelöscht werden, bleiben die Datenträger im MAREN-Katalog weiterhin eingetragen.
- Im Dateikatalog TSOSCAT können beliebig viele Einträge mit derselben VSN existieren, im MAREN-Katalog muss die VSN eindeutig sein.
- Im MAREN-Katalog können beliebig viele Einträge mit demselben Dateinamen existieren, im TSOSCAT ist der Dateiname eindeutig.

### Freibandzuweisung und automatische Initialisierung

Eine wichtige Komponente des MAREN-Systems ist die so genannte automatische Freibandzuweisung MARENUCP. Damit können z.B. im Fall einer unspezifizierten Bandanforderung („SCRATCH"-Bänder) freie Bänder aus dem MAREN-Datenträgerpool automatisch zugewiesen werden. Dabei werden gleichzeitig die verarbeiteten Bänder im MAREN-Katalog als reserviert gekennzeichnet.
MARENUCP prüft, ob ein Band, das angefordert und vom Operator montiert wurde, initialisiert werden muss. Falls notwendig, wird zuerst die Initialisierung durchgeführt und anschließend die Reservierung und Zuweisung vorgenommen.

### Schnittstellen zur Bandverarbeitung

Bei der Magnetband-Reservierung und -Verarbeitung über das BS2000-DVS wird automatisch das MAREN-System für folgende Aufgaben aktiv:

- Kontrolle der Zugriffsberechtigung
- Prüfung auf Verfügbarkeit
- Aktualisierung der Archiveinträge
- Ausgabe der Transportmeldungen auf Bedienstation
- Einsetzen der DEVICE-Parameter

Der Aufruf dieser Prüfroutinen kann von folgenden Ereignissen veranlasst werden:

- Band belegen
- TFT-Eintrag erstellen
- Datei eröffnen
- Spulen wechseln
- Datei schließen
- Band freigeben

### Kontrolle der Zugriffsberechtigung

Vor jeder Verarbeitung eines Bandes kontrolliert MAREN, ob der Aufrufer berechtigt ist, auf den Datenträger zuzugreifen. Folgende Bedingungen müssen vor jedem Zugriff erfüllt sein:

- Falls das Band einer fremden Benutzerkennung zugeordnet ist, muss es im MAREN-Katalog das Attribut USER-ACCESS=*ALL-USERS besitzen
  (Ausnahme: Aufrufer unter TSOS).
- Falls das Band durch ein Datenträgerkennwort geschützt ist, muss dieses Kennwort angegeben worden sein.

- Falls das Band mit einem ARCHIVE-Directory in Verbindung steht, dann darf auf dieses Band nur mit ARCHIVE zugegriffen werden.
- Bei Ausgabe auf Band muss der schreibende Zugriff erlaubt sein; das Attribut FOREIGN-READ-ONLY darf im MAREN-Katalog nicht hinterlegt sein (Ausnahme: Aufrufer unter TSOS).
- Bei Ausgabe auf Band muss das Dateifreigabedatum kleiner oder gleich dem Tagesdatum sein.
- Falls MAREN-RZ-Exitroutinen im Einsatz sind, müssen diese den Zugriff auf das Band erlauben.

Ist eine dieser Bedingungen nicht erfüllt, wird der Zugriff auf den betreffenden Datenträger abgelehnt und eine entsprechende Meldung an die Datensichtstation geschickt.

#### Prüfung auf Verfügbarkeit

Vor jedem Zugriff auf ein Band prüft MAREN, ob die folgenden Voraussetzungen für die Verfügbarkeit des Datenträgers erfüllt sind:

- Die Archivnummer des Bandes muss im MAREN-Katalog eingetragen sein.
- Die Archivnummer muss laut Attribut DEVICE-TYPE ein Band bezeichnen.
- Das Band darf nicht ausgeliehen sein.
- Das Band muss für den Aufrufer reserviert sein.
- Das Band muss lokal verfügbar sein; d.h der in einem speziellen Feld des Archiveintrags enthaltene Lagerort darf in der so genannten Lagerorttabelle nicht als „entfernt“ gekennzeichnet sein.
- Das Band darf nicht gerade an einem anderen Server verarbeitet werden (die Prüfung auf eine gerade laufende Verarbeitung am eigenen Server wird nicht von MAREN, sondern von BS2000/OSD durchgeführt).
- Beim Verarbeitungsmodus INPUT (außer bei Zugriff über ARCHIVE) müssen die Dateinamen im MAREN-Archiveintrag und in der TFT übereinstimmen. Dabei werden die Namensteile Katalogkennung, Benutzerkennung und Version ignoriert.

Installation und Einsatz von MAREN im Data Center sind in den Handbüchern zu „MAREN“ [31] ausführlich beschrieben.

### 5.9.2 Einsatzmöglichkeiten privater Platten

In BS2000/OSD können Platten grundsätzlich als private Platte oder als gemeinschaftliche Platte genutzt werden.

Vor dem ersten Einsatz müssen die Datenträger mit dem Dienstprogramm VOLIN initialisiert werden. Dabei wird u. a. das maximal 6-stellige Datenträgerkennzeichen (Archivnummer) in den entsprechenden Kennsatz (Standard Volume Label) geschrieben. Anhand dieses Datenträgerkennzeichens (Volume Serial Number, VSN) können die privaten und gemeinschaftlichen Platten unterschieden werden.
Die VSN einer privaten Platte ist, mit der Einschränkung, dass sie nicht mit der einer gemeinschaftlichen Platte übereinstimmen darf, frei wählbar. Die Regeln und Möglichkeiten zur Vergabe einer VSN für gemeinschaftliche Platten sind im Abschnitt „VSN und Pubset-Adressierung“ auf Seite 316 erläutert.

Für private Platten gibt es zwei Anwendungsmöglichkeiten:

- DVS-Anwendungen
  Die Platte ist durch eine oder mehrere DVS-Anwendungen belegt.
  Dabei ist der Taskbelegungsmodus standardmäßig task-shareable. Eine task-exklusive Belegung erreicht man mit dem Kommando SECURE-RESOURCE-ALLOCATION. Der Systembelegungsmodus richtet sich nach den Generierungseigenschaften des Plattengerätes, in dem die Platte montiert ist. Mit dem Operatorkommando SET-DISK-PARAMETER kann der Systembelegungsmodus generierungsunabhängig und plattenspezifisch eingestellt werden.

- Sonderanwendungen (SPECIAL-Anwendungen)
  Die Platte ist durch eine privilegierte Anwendung – z.B. VOLIN, FDDRL – belegt.
  Die SPECIAL-Anwendungen sind immer task- und system-exklusive Belegungen. Die Belegungsdauer wird von der SPECIAL-Anwendung bestimmt, sie kann vom Operator nicht beeinflusst werden.

#### Shareable Private Disk (SPD)

Ist ein Plattengerät über direkte Hardware-Verbindungen von mehr als einem Server erreichbar, dann kann in diesem Laufwerk eine Platte als gemeinsam benutzbare private Platte (SPD) zur Verfügung gestellt werden.

Das Datenverwaltungssystem von BS2000/OSD koordiniert den Zugriff zu den gemeinsam benutzbaren Plattengeräten und synchronisiert den Zugriff zu den Dateien auf den privaten Datenträgern.

Ein Plattengerät wird als gemeinsam benutzbar (SPD-fähig) bezeichnet, wenn es mindestens 2 Zugriffswege von verschiedenen Servern gibt, über die auf das Gerät zugegriffen werden kann.

Der hier verwendete Begriff „gemeinsam benutzbar“ ist eine Eigenschaft des Plattengeräts und nicht zu verwechseln mit der Eigenschaft „shareable“ für Dateien oder Datenträger. Die Shareable-Funktion für Plattengeräte ist festgelegt durch die vorgegebene Hardware-Konfiguration.
Ein SPD-fähiges Plattengerät verliert diese Eigenschaft nur, wenn es bei einer Hardware-Rekonfiguration als „nicht gemeinsam benutzbar“ definiert wird.

SPD-Platten werden nur am Typ S-Kanal der S-Server bedient.

Einzelheiten zum Einsatz von Shareable Private Disks im Rechnerverbund sind im Handbuch „HIPLEX MSCF“ [33] beschrieben.

### Dual Recording by Volume (DRV)

Mit dem Subsystem DRV können neben gemeinschaftlichen Platten auch Privatplatten, die nicht mit der Funktion SPD belegt sind, im System doppelt geführt werden.
Shared-Pubsets können nicht für DRV verwendet werden.
Jeder Schreibauftrag des DVS wird doppelt ausgeführt und jeder Leseauftrag wird auf der Platte mit der jeweils geringeren Auslastung abgewickelt.

*Voraussetzungen*

Um zwei Platten als eine logische Platte betreiben zu können, müssen folgende Bedingungen erfüllt sein:

- gleicher Gerätetyp
- gleicher Steuerungstyp
- gleicher Kanaltyp
- gleicher logischer Aufbau durch gleiche VOLIN-Parameter
- gleiche VSN
- gleicher Timestamp
- homogener Eintrag für das DRV-Aufzeichnungsverfahren

*Bedienung und Steuerung*

Die gesamte Funktionalität von DRV ist im Ein-/Ausgabe-System, im NDM sowie im Subsystem DRV realisiert und muss weder vom DVS noch von Benutzerprogrammen berücksichtigt werden.
Der DRV-Betrieb wird ausschließlich von der Systembetreuung eingeleitet, gesteuert, überwacht und beendet.

Das Produkt DRV ist im Handbuch „DRV“ [17] ausführlich beschrieben.

## 5.10 IORM: Steuerung von I/O-Ressourcen

Das Dienstprogramm IORM verbessert die I/O-Eigenschaften von BS2000/OSD im Native- und im VM2000-Betrieb.

In IORM (und dem IORM-Subsystem) sind folgende Funktionen zur autonomen, dynamischen Steuerung der I/O-Betriebsmittel Kanal, Steuerung, Pfad und Gerät realisiert:

- IOPT: I/O-Prioritäten-Steuerung für Tasks (I/O Priority Handling for Tasks)
- DPAV (S-Server):
  Dynamische I/O-Lastverteilung für Platten (Dynamic Parallel Access Volume)
- DDAL: Optimierte Lastverteilung im Betrieb mit ETERNUS CS HE
  (Dynamic Device Allocation)
- TCOM: Kompression im LTO-Gerät anpassen (Dynamic Tape Compression)
- IOLVM (S-Server):
  Begrenzung der I/O-Aufnahme einzelner VM2000-Gastsysteme
  (I/O Limit for Virtual Machines)

IORM sammelt im laufenden Betrieb Daten über die Auslastung der I/O-Betriebsmittel und steuert den I/O-Betrieb abhängig von vorgegebenen Schwellenwerten.

Die IORM-Funktionen IOPT, DPAV und IOLVM steuern Plattengeräte.
Die IORM-Funktionen DDAL und TCOM steuern Bandgeräte.

Wenn im VM2000-Betrieb IORM im Monitorsystem und in den beteiligten BS2000-Gastsystemen im Einsatz ist, dann tauschen die IORM-Subsysteme über eine interne Schnittstelle I/O-Daten und Kontrollinformationen aus.

IORM kann im Native-Betrieb und unter VM2000 eingesetzt werden.
IORM arbeitet aber nicht Server-übergreifend.

Eine detaillierte Beschreibung von IORM finden Sie im Handbuch „Dienstprogramme“ [15].

## 5.11 SANCHECK: Überprüfung der SAN-Konfiguration

Die BS2000/OSD-Server werden mit modernen Speichersystemen zunehmend über die FibreChannel-Technik verbunden (für S-Server über den Typ FC). Dabei werden die Speichersysteme in der Regel nicht direkt an den FibreChannel-Adapter eines Servers angeschlossen, sondern über einen **Switch**. Ein solcher FC-Switch ermöglicht gleichzeitig mehrere Verbindungen zwischen den Geräten, die an seinen Ports angeschlossen sind. An einen FC-Switch können über spezielle Anschlüsse wiederum andere Switches angeschlossen werden. Ein Netz, das aus einem oder mehreren FC-Switches gebildet wird, heißt Fabric. Ein Netz aus mehreren Speichersystemen, die mit FC-Switches verbunden sind, wird als Storage Area Network (SAN) bezeichnet.

Aus Sicht von BS2000/OSD sind die FC-Switches transparent. BS2000/OSD verwendet die Steuerungen und Geräte, die über FibreChannel (für S-Server über den Typ FC-Kanal) angeschlossen sind, ohne Informationen über die Verbindungen in der Fabric zu haben.

Wenn es zu Problemen beim Zuschalten von Geräten oder zu Fehlern im laufenden Betrieb kommt, ist es oft schwer, die Ursache hierfür zu erkennen. Eine INOP- oder NINT-Meldung der Gerätefehlerbehandlung kann durch Verbindungsstörungen an beliebiger Stelle im SAN hervorgerufen werden. Möglicherweise kann ein Gerät gar nicht erst zugeschaltet werden, weil die in BS2000/OSD generierten Pfade oder WWPNs (World Wide Port Number) physikalisch nicht vorhanden sind oder weil die generierten Verbindungen zwischen Kanal und Steuerung in den Switches nicht zugelassen werden.

Das Dienstprogramm SANCHECK bietet für diese Fälle komfortable Diagnosehilfsmittel. Dabei werden Hilfen für zwei Problembereiche geboten:

- Erkennung von Generierungsfehlern (nur relevant bei S-Servern)
- Lokalisierung von Fehlerzuständen im SAN

Mit der Anweisung SHOW-SAN-PATH können gezielt Verbindungswege durch die Fabric(s) des SAN zwischen vorgegebenen Hardware-Einheiten (Kanälen, Steuerungen) gesucht und ihr Zustand geprüft werden. Bei Angabe des Operanden INFORMATION= *ERROR wird über die Meldungen `SAN0Pnn` gezielt angegeben, wo es auf den Verbindungswegen im SAN für die generierten IO-Pfade Probleme gibt.

Mit der Anweisung SHOW-SAN-CONFIGURATION können gezielt Informationen über die Fabrics, Switches und Ports abgerufen werden. Die Verbindungen der Switches innerhalb einer Fabric werden aufgezeigt. Für alle Ports der Switches werden deren Verbindungen („Link-Neighbours“) und die Zustände der jeweiligen Einheiten angegeben.

Für ausführliche Informationen zu SANCHECK siehe Handbuch „Dienstprogramme“ [15].

# 6 BS2000-Benutzerverwaltung

In diesem Kapitel werden Aufbau und Verwaltung des Benutzerkatalogs beschrieben.
Jeder Pubset enthält:

- den eigenen MRSCAT-Eintrag (gespeichert im TSOSCAT)
- einen Benutzerkatalog SYSSRPM
- einen Dateikatalog TSOSCAT

Die Adressierung der Pubsets erfolgt über das Katalogverzeichnis MRSCAT. Obwohl jeder Pubset ein solches Katalogverzeichnis haben kann, wird im Systemlauf nur der MRSCAT des Home-Pubsets zur Adressierung der Pubsets herangezogen.

Auf jedem Pubset existieren unter der Kennung TSOS die Dateien SYSSRPM und SYSSRPM.BACKUP (auch .BACKUP genannt). Beide Dateien werden beim Importieren eines Pubsets von einer Systemtask geöffnet und von dieser beim Export wieder geschlossen. Damit ist ein direkter Zugriff auf die Dateien nicht möglich.

**Benutzerkatalog**

Die Datei SYSSRPM stellt den Benutzerkatalog dar.
Die Datei .BACKUP ist eine Kopie des Benutzerkatalogs. Sie fungiert als Hilfsdatei, wenn der Benutzerkatalog gesichert oder mit Hilfe der Sicherung rekonstruiert werden soll.

Der Benutzerkatalog wird mit HSMS oder ARCHIVE gesichert. Beim Sichern und Wiedereinspielen der Benutzerdaten sind zu unterscheiden:

- Sichern des aktuellen Benutzerkataloges eines Pubsets
- Wiedereinspielung eines gesicherten Benutzerkataloges in die Datei .BACKUP
- Rekonstruktion der Datei SYSSRPM mit Hilfe des wiedereingespielten Benutzerkataloges, sofern vorher eine Wiedereinspielung erfolgreich abgeschlossen werden konnte.

Sicherung und Wiedereinspielung erfolgen im laufenden System von einem bzw. auf einen importierten Pubset. Die Rekonstruktion erfolgt ausschließlich während des Imports eines Pubsets.

Die Sicherung von Benutzerkatalogen ist für SF- wie SM-Pubsets gleich.
Die Benutzerkataloge von SF- und SM-Pubsets unterscheiden sich im Prinzip nicht.
Zu Besonderheiten des Benutzerkatalogs in Bezug auf das SMS-Konzept siehe Abschnitt Abschnitt „Benutzerkatalog und SMS-Konzept“ auf Seite 271.

### Die Datei $TSOS.SYSSRPM

Die Systemdatei $TSOS.SYSSRPM ist eine NK-ISAM-Datei, in der alle Informationen zu Benutzerkennungen von BS2000/OSD aufbewahrt werden.

Die Datei $TSOS.SYSSRPM existiert auf jedem Pubset. Auf sie wird aus einer Server-Task zugegriffen (SRPM-Tasks TSN RP01 oder RP02). Mit dem Import wird die Datei durch die Server-Task eröffnet und bleibt bis zum Export im geöffneten Zustand. Ein direkter Zugriff auf $TSOS.SYSSRPM ist nicht möglich.

Die Rekonstruktion des Benutzerkatalogs, also die Übernahme des Inhalts der Datei .BACKUP in die Datei SYSSRPM, erfolgt nur während eines Imports; ARCHIVE ist daran nicht mehr beteiligt.

| **Kommando** | **Bedeutung** |
|---|---|
| ADD-USER | Eintrag im Benutzerkatalog erstellen |
| LOCK-USER | Zugang zum System sperren |
| MODIFY-USER-ATTRIBUTES | Eintrag im Benutzerkatalog ändern |
| MODIFY-USER-PUBSET-ATTRIBUTES | Pubset-spezifische Benutzerattribute einer Benutzerkennung ändern |
| REMOVE-USER | Eintrag eines Benutzers im Benutzerkatalog löschen |
| SHOW-USER-ATTRIBUTES | Informationen aus dem Benutzerkatalog anfordern |
| UNLOCK-USER | Zugangssperre zum System aufheben |
| **Makro** | **Bedeutung** |
| RDUID | Benutzerkennung abfragen |
| SRMUINF | Daten aus dem Benutzerkatalog in einen Bereich übertragen |

Tabelle 16: Schnittstellenübersicht zur Verwaltung des Benutzerkataloges

## 6.1 Struktur eines Benutzerkatalogs

Die Datei $TSOS.SYSSRPM enthält Folgendes:

- Informationen zur Zugangskontrolle für Benutzerkennungen (auch bei erweitertem Zugangsschutz mit SECOS)
- Systemglobale Privilegien
- BS2000-Benutzergruppen

Die Datei SYSSRPM stellt den Benutzerkatalog dar.

Der Benutzerkatalog des Home-Pubsets enthält **system-spezifische** Daten, wie z.B. Zugriffsrechte zu BS2000/OSD (Benutzerkennung, Abrechnungsnummer, Kennwort der Benutzerkennung, SPOOLOUT-Klasse, Versandanschrift usw.) und die Standardkatalog-Kennung.

In den Benutzerkatalogen der importierten Pubsets sind **pubset-spezifische** Daten verankert. Sie enthalten u.a. Informationen, die den einzelnen Benutzer auf dem jeweiligen Pubset betreffen, z.B.:

- den maximal zur Verfügung stehenden Speicherplatz
- ggf. die Erlaubnis, diese Grenze zu überschreiten
- Katalogkontingente

Die system-spezifischen Informationen des Home-Pubsets können auch in den Benutzerkatalogen der importierten Pubsets vorhanden sein; für Überprüfungen wird allerdings nur der Benutzerkatalog des Home-Pubsets vom System herangezogen. Die system-spezifischen Daten auf den importierten Pubsets werden nur dann benötigt, wenn diese Pubsets als Reserve-Home-Pubsets zur Systemeinleitung von BS2000/OSD und für den Systemlauf benutzt werden.

## 6.2 Führen des Benutzerkatalogs

Der Benutzerkatalog wird beim erstmaligen Importieren eines Pubsets (nach dessen Erstellung mit SIR) mit dem Kommando IMPORT-PUBSET ACTUAL-JOIN=*FIRST automatisch eingerichtet. Er enthält dann standardmäßig folgende Benutzerkennungen:

| | |
|---|---|
| TSOS | Benutzerkennung der Systemverwaltung |
| SYSPRIV | Benutzerkennung für die Vergabe von Privilegien bei Einsatz des Software-Produkts SECOS |
| SYSDUMP | Benutzerkennung, unter der Systemdumps abgelegt werden |
| SYSOPR | Benutzerkennung für das Operating |
| SYSSPOOL | Benutzerkennung für die Spoolverwaltung |
| SERVICE | Benutzerkennung für den Service. Unter dieser Benutzerkennung werden vom HW- und SW-Service spezielle Programme zur Betriebssicherung eingesetzt |
| SYSGEN | Benutzerkennung für die Hardware-Generierung |
| SYSHSMS | Benutzerkennung für das HSMS-Datenarchiv |
| SYSSNAP | Benutzerkennung, unter der SNAP-Dumps abgelegt werden |
| SYSUSER | Benutzerkennung für Userdumps, die nicht unter der Benutzerkennung des Verursachers abgelegt werden können oder sollen |
| SYSAUDIT | Benutzerkennung für REPLOG-Verwaltung sowie für die SAT-Auswertung und SAT-Datei-Verwaltung bei Einsatz von SECOS |
| SYSNAC | Benutzerkennung für Network Administration Center (Netzverwaltung) |
| SYSROOT | Benutzerkennung für die POSIX-Verwaltung |
| SYSSNS | Benutzerkennung für den SPOOL Notification Service |
| SYSMAREN | Benutzerkennung für die MAREN-Verwaltung |
| SYSSOPT | Benutzerkennung für das Programm SPACEOPT |



**ACHTUNG!**
Wird ein schon einmal importierter Pubset erneut mit ACTUAL-JOIN=*FIRST importiert, wird nicht nur der Benutzerkatalog auf seine Standardbelegung zurückgesetzt, sondern auch alle auf ihm befindlichen Daten – außer Dateien unter $TSOS – werden gelöscht.

Die Benutzerkennungen des Systems, mit Ausnahme von TSOS und SERVICE, werden bei der Initialisierung gesperrt gesetzt und können mit dem Kommando UNLOCK-USER freigegeben werden.

Für jeden Benutzer, der Zugriff zum System haben soll, muss die Systembetreuung einen Eintrag im Benutzerkatalog des Home-Pubsets und dem des zugewiesenen Standard-Pubsets erstellen. Für die LOGON-Validierung ist der Eintrag im Benutzerkatalog des Home-Pubsets ausschlaggebend. In Absprache mit dem Benutzer werden ihm, neben den Daten zu seiner Identifizierung, bestimmte Betriebsmittel und Berechtigungen zugeordnet:

**Identifizierung**: Benutzerkennung, Kennwort, SPOOLOUT-Klasse, Versandanschrift

**Betriebsmittel**: Standard-Pubset, Benutzeradressraum, Speicherplatz auf gemeinschaftlichen Platten, CPU-Zeit, Meldungssprache, Zuordnung einer Gruppensyntaxdatei

**Berechtigungen**: Überschreitung des zugewiesenen Speicherplatzes, Nutzung bestimmter Task-Attribute, Ablaufpriorität, Testprivilegien, Nutzung von Hardware- und Linkage-AUDIT, Nutzung von Net-Storage

Die Kommandos ADD-USER und MODIFY-USER-ATTRIBUTES dienen zum Erstellen bzw. Aktualisieren von Einträgen im Benutzerkatalog.

Mit dem Kommando SHOW-USER-ATTRIBUTES kann sich die Systembetreuung die Benutzereinträge ausgeben lassen und erhält auf diese Art und Weise einen Überblick über den Inhalt des Benutzerkataloges. Ferner kann die Systembetreuung mit den Kommandos LOCK-USER und REMOVE-USER Benutzerkennungen sperren bzw. löschen und mit UNLOCK-USER wieder entsperren.

Die System-Benutzerkennungen können nicht gelöscht werden (Ausnahme: die Benutzerkennung SERVICE bei Einsatz des Produkts SECOS, siehe das SECOS-Handbuch „Zugangs- und Zugriffskontrolle“ [48]) .

# 6.3 Sicherungskonzept für Benutzerkataloge

## 6.3.1 Sichern des aktuellen Benutzerkatalogs eines Pubsets

Zur Sicherung des Benutzerkataloges muss die Datei $TSOS.SYSSRPM.BACKUP mit dem Subsystem ARCHIVE (Anweisung SAVE) oder HSMS (Anweisung BACKUP-FILES) gesichert werden. Diese Sicherung erfolgt für den Benutzer im Wesentlichen wie die Sicherung jeder anderen Datei, wobei der Dateiname nur als Platzhalter fungiert.
Die aktuellen Daten werden während des SAVE-Laufs von der Benutzerverwaltung an ARCHIVE/HSMS übergeben, die eigentliche .BACKUP-Datei behält ihre Platzhalterfunktion und bleibt leer.
Um die gesicherten Daten wirksam werden zu lassen, muss die .BACKUP-Datei vor IPL bzw. IMCAT von der Sicherung eingelesen (mit RESTORE/RESTORE-FILES) und der IPL bzw. IMCAT entsprechend parametrisiert werden.

Wird eine Differenzsicherung der Datei gefordert (Operation SAVE ...,CHANGED=YES), so erfolgt diese in jedem Fall, da die Datei nach der letzten Bearbeitung durch ARCHIVE von der sie verwaltenden Systemtask wieder geöffnet wurde.

Ein Export für die .BACKUP-Datei wird abgewiesen.



Da bei der Rekonstruktion eine Manipulation des Inhalts des gesicherten Benutzerkataloges nicht festgestellt werden kann, muss die Datei von der Systembetreuung durch geeignete organisatorische Maßnahmen geschützt werden.

## 6.3.2 Wiedereinspielen eines gesicherten Benutzerkatalogs

Zum Wiedereinspielen des Benutzerkataloges muss die Datei $TSOS.SYSSRPM.BACKUP mit Hilfe des Subsystems ARCHIVE (Operation RESTORE) wiedereingespielt werden.
Sofern bei der Sicherung der Datei $TSOS.SYSSRPM.BACKUP diese umbenannt wurde, muss diese Umbenennung bei der Wiedereinspielung rückgängig gemacht werden. Da die Datei permanent existiert, muss bei der Operation RESTORE der Operand REPLACE auf einen der Werte YES, ALL oder ALLP gesetzt werden.



Es ist zulässig, einen gesicherten Benutzerkatalog auf einem anderen Pubset als demjenigen, von dem gesichert wurde, einzuspielen. Zu beachten ist dabei, dass Attribute, wie z.B. der Standard-Pubset, manuell aktualisiert werden müssen.

### 6.3.3 Rekonstruktion der Datei SYSSRPM

Die Rekonstruktion des Benutzerkataloges kann beim Startup (im Falle des Home-Pubsets) oder mit dem Kommando IMPORT-PUBSET veranlasst werden. Voraussetzung ist die erfolgreiche Wiedereinspielung eines Benutzerkataloges in die Datei .BACKUP vor dem letzten Shutdown bzw. Export.

Wird ein Pubset neu eingerichtet, kann die Rekonstruktion der Datei SYSSRPM auch durch SIR erfolgen.

Eine Rekonstruktion betrifft nicht nur den Benutzerkatalog, sondern auch den Dateikatalog TSOSCAT. Der Dateikatalog führt eine eigene Liste von Benutzerkennungen, die der Abbildung Benutzerkennung zu PBN (Primary Block Number, kennzeichnet den Eigentümer der Datei) dient. Die Rekonstruktion muss die Konsistenz beider Benutzerstrukturen wahren. Die Benutzerstruktur des vorhandenen Benutzerkataloges SYSSRPM wird nicht in den Rekonstruktionsprozess miteinbezogen.

Ziel ist es, die beiden Benutzerstrukturen von .BACKUP und TSOSCAT zu vereinigen und dabei mögliche Inkonsistenzen zu beseitigen.

*Beispiel*

Menge A: Menge aller Benutzer, die in der Datei .BACKUP existieren, jedoch nicht im TSOSCAT

Menge B: Menge aller Benutzer, die sowohl in der Datei .BACKUP also auch im TSOSCAT existieren

Menge C: Menge aller Benutzer, die im TSOSCAT existieren, jedoch nicht in der Datei .BACKUP

Im Idealfall sollten die Mengen A und C leer sein, da sonst bei der Rekonstruktion entweder Benutzerattribute oder Dateien verloren gehen können.

Die Rekonstruktion wird durch den Systemparameter RECONUC gesteuert. Er kann über den Startup-Parameterservice eingestellt und geändert werden. Beim DIALOG-Startup besteht eine zusätzliche Möglichkeit, RECONUC zu ändern: Über die Meldung `NSI6010` werden der voreingestellte Wert für RECONUC ausgegeben und Änderungswünsche abgefragt.

Zu beachten ist, dass die Systemparameter RECONUC und STUPTYPE in Zusammenhang stehen. Sofern STUPTYPE=J oder T gewählt wurde, wird zuerst der First-Startup (mit/ohne Rücksetzen des Benutzerkatalogs) ausgeführt und anschließend der Wert für RECONUC ausgewertet. Die Meldung `NSI6220` informiert über einen ungültigen Wert für RECONUC oder STUPTYPE und das Setzen auf einen Standardwert, der in der Meldung angegeben wird.

RECONUC kann die Werte N, B, T, A oder R annehmen. In der folgenden Aufzählung wird der entsprechende Wert beim Kommando IMPORT-PUBSET in eckigen Klammern angegeben.

1. Keine Rekonstruktion (≙ RECONUC=N; [NO])

   Der Import (bzw. Startup) erfolgt ohne eine Rekonstruktion des Benutzerkataloges.

2. Rekonstruktion mittels BACKUP (≙ RECONUC=B; [SCOPE BACKUP])

   Ein neuer Benutzerkatalog wird mittels Inhalt der Datei .BACKUP aufgebaut.
   Alle Benutzer und ihre Attribute, die in der .BACKUP-Datei gesichert wurden, werden rekonstruiert. Dateien und Jobvariablen, die keinem der Benutzer gehören, die in der .BACKUP-Datei eingetragen sind, werden gelöscht.

   Bezogen auf die Mengen A, B und C bedeutet das:

   – Für die Menge A aller Benutzerkennungen:
     Neueinrichtung mit den gesicherten Attributen.
   – Für die Menge B aller Benutzerkennungen:
     Update der vorhandenen mit den gesicherten Benutzerattributen.
   – Für die Menge C aller Benutzerkennungen:
     Löschen ihrer Dateien und Jobvariablen.

   Dieser Modus ist bei regelmäßiger Sicherung zu empfehlen.

3. Rekonstruktion mittels TSOSCAT (≙ RECONUC=T; [SCOPE TSOSCAT])

   Ein neuer Benutzerkatalog wird erzeugt, in den ausschließlich die Benutzer eingetragen werden, die auch in der Datei $TSOS.TSOSCAT einen Eintrag besitzen. Sofern vorhanden, werden deren Benutzerattribute der Datei .BACKUP entnommen, andernfalls wird ein Standardbenutzereintrag erstellt.

   Bezogen auf die Mengen A, B und C bedeutet das:

   – Für die Menge A aller Benutzerkennungen:
     keine Übernahme in die rekonstruierte Benutzerstruktur. Dadurch kann die Privilegienverteilung oder die Gruppenstruktur auf dem betreffenden Pubset dahingehend gestört werden, dass z.B. ein Benutzer, der als einziger in der .BACKUP-Datei ein bestimmtes Privileg besaß, nicht in den neuen Benutzerkatalog übernommen wird oder keine Übernahme eines Gruppenverwalters erfolgt.
   – Für die Menge B aller Benutzerkennungen:
     Update der vorhandenen mit den gesicherten Benutzerattributen.
   – Für die Menge C aller Benutzerkennungen:
     Einrichtung mit Standard-Attributen und den Erhalt der Dateien, Jobvariablen und Guards.

   Dieser Modus ist dann zu empfehlen, wenn der Erhalt von Dateien im Vordergrund steht und Benutzerkennungen, soweit sie zum Zeitpunkt der Sicherung bereits existierten, rekonstruiert werden sollen.

4. Rekonstruktion mittels BACKUP und TSOSCAT (≙ RECONUC=A; [SCOPE ALL])

   Ein neuer Benutzerkatalog wird mittels Inhalt der Dateien $TSOS.SYSSRPM.BACKUP und $TSOS.TSOSCAT aufgebaut; er enthält also Einträge für alle Benutzer, die in mindestens einer der beiden Dateien verzeichnet sind. Sofern vorhanden, werden die Benutzerattribute der Datei .BACKUP entnommen, andernfalls wird ein Standardbenutzereintrag erstellt und die betroffenen Kennungen gesperrt.

   Analog zu Scope BACKUP wird die Benutzerstruktur zum Zeitpunkt der Sicherung wiederhergestellt.

   Analog zu Scope TSOSCAT bleiben Dateien von Benutzerkennungen, die nach erstellen der Sicherung eingerichtet wurden, durch wiedereinrichten der Benutzerkennungen mit Standardattributen erhalten. Im Falle zweier großer, disjunkter Benutzerstrukturen kann dies das Fassungsvermögen des TSOSCAT mit maximal 8189 Benutzerkennungen sprengen. In diesem Fall wird der Import sofort nach Festellung dieser Tatsache abgebrochen, und der Import kann mit einem der Modi BACKUP oder TSOSCAT wiederholt werden.

5. Rücksetzen des Benutzerkataloges (≙ RECONUC=R; [RESET])

   Bei dieser Strategie, die bislang nur als Antwort auf die Meldung `SRM2012` zur Verfügung stand, wird der Benutzerkatalog allein auf Basis der Benutzerstruktur des TSOSCAT wiederhergestellt, eine .BACKUP-Datei wird nicht benötigt. Alle Benutzer erhalten Standardattribute und alle Kennungen außer TSOS und SERVICE werden gesperrt.

   Diese Funktion erlaubt der Systembetreuung die formale Wiederherstellung des Benutzerkataloges unter Erhalt der Dateien. Die inhaltliche Wiederherstellung (also die Wiederherstellung der Benutzerattribute) muss in einem zweiten Schritt mit Hilfe der Rekonstruktion erfolgen.

   Als Alternative im Fehlerfall bietet sich nur noch der First-Start an, bei dem alle Kennungen außer denen der Systemverwaltung und alle Dateien, die nicht der Kennung TSOS gehören, verlorengehen.
   Wurde die SYSSRPM durch einen Systemfehler zerstört, sollte man die Strategie der vollständigen Pubset-Rekonstruktion wählen, denn dieser Systemfehler könnte auch andere Dateien zerstört oder beschädigt haben.

Den ordnungsgemäßen Verlauf der Rekonstruktion kann der Operator anhand von zwei Meldungen verfolgen, von denen die erste zu Beginn die Rekonstruktionsbasis über die Katalogkennung und den Sicherungszeitpunkt dokumentiert (`SRM2017` bei Rekonstruktion mit (*BY-BACKUP) bzw. `SRM2018` bei Rekonstruktion ohne .BACKUP-Datei (*RESET)) und die zweite am Ende die Anzahl der rekonstruierten Benutzerkennungen angibt (`SRM2019` bei Rekonstruktion mit (*BY-BACKUP) bzw. `SRM2020` bei Rekonstruktion ohne .BACKUP-Datei (*RESET)).

Dem Systembetreuer wird darüber hinaus eine Logging-Datei zur Verfügung gestellt, die über Folgendes Auskunft gibt (SYS.SRPM.RECON.LOG.<datum.uhrzeit>):

- die relevanten Attribute des rekonstruierten Pubsets und der Rekonstruktionsbasis
- jede beteiligte Benutzerkennung, deren Rekonstruktion oder Löschung
- die durchgeführten Anpassungen der Rekonstruktionsbasis an die aktuelle System-umgebung

Soll die Rekonstruktion der Wiederherstellung eines defekten Benutzerkatalogs dienen, der sich aber als Folge des Defekts nicht importieren lässt, muss zunächst eine rudimentäre Korrektur über die Funktion RESET erfolgen, bei der auf Basis der in TSOSCAT bekannten Benutzerkennungen ein neuer Benutzerkatalog aufgebaut wird. Die einzelnen Benutzer-kennungen erhalten Standard-Attribute und sind alle bis auf TSOS und SERVICE gesperrt. Im Gegensatz zum First-Start bleiben auf diese Weise alle Dateien erhalten und der voll-ständige Neuaufbau des Pubsets kann vermieden werden.
Sofern kein Zip-Import gefordert ist, wird der defekte Benutzerkatalog nicht gelöscht, son-dern für eine spätere Diagnose des Problems, das zum Reset geführt hat, unter dem Namen :catid:$TSOS.SYS.SRPM.RECON.DIAG.<datum.uhrzeit> gesichert.



**ACHTUNG!**
Bei der Weitergabe eines defekten Benutzerkatalogs an Dritte ist zu bedenken, dass eine unberechtigte Rekonstruktion der Benutzerdaten nicht ausgeschlossen werden kann.

Die Funktion RESET kann neben der Angabe als Rekonstruktions-Typ (siehe oben) auch als Antwort auf die Meldung `SRM2012` angefordert werden. Diese Meldung wird nur auf Grund von Fehlern in der Benutzerverwaltung angeboten, Fehler in anderen Komponenten wie z.B. der Gruppenverwaltung, führen zum Abbruch der IMPORT-PUBSET-Verarbeitung.

### Anpassung an die Systemumgebung

Im Idealfall sollte ein Pubset mit seiner eigenen Sicherung rekonstruiert werden. Ist dies nicht der Fall, können Inkonsistenzen zwischen den Systemumgebungen zu den Zeitpunk-ten der Sicherung und Rekonstruktion bestehen.

### Standard-Pubset (Default-Pubset)

Das Attribut DEFAULT-PUBSET aller System-Benutzerkennungen erhält die Catid des rekonstruierten Pubsets. Da alle System-Benutzerkennungen anläßlich ihrer Einrichtung beim First-Start der Benutzerverwaltung in diesem Punkt gleich behandelt werden, ist dies auch bei der Rekonstruktion der Fall.

Das Attribut DEFAULT-PUBSET der übrigen Benutzerkennungen bleibt unberührt. Wurde die Sicherung einer inkompatiblen Pubset-Konfiguration rekonstruiert, muss die System-verwaltung in eigener Verantwortung die lokal gültigen Default-Pubsets zuweisen.

### Logon-Kennwörter

Wurde der Benutzerkatalog eines Systems ohne Kennwortverschlüsselung (Systemparameter ENCRYPT) gesichert und auf einem System mit Kennwortverschlüsselung rekonstruiert, werden alle Logon-Kennwörter verschlüsselt.
Im umgekehrten Fall ist eine entsprechende Maßnahme unmöglich!

Durch die Rekonstruktion erhalten alle Benutzerkennungen ihre Logon-Kennwörter zum Zeitpunkt der Sicherung zurück. Es liegt in der Verantwortung jedes einzelnen Benutzers, sich an dieses Kennwort zu erinnern. Dies gilt insbesondere für TSOS oder eine andere Benutzerkennung mit dem Privileg USER-ADMINISTRATION.

### SECOS-Attribute

Wurde der Benutzerkatalog eines Systems mit installiertem SECOS gesichert und auf einem System ohne SECOS rekonstruiert, werden die Einstellungen der Logon-Parameter und der Privilegien aller Benutzerkennungen auf die Standard-Einstellungen zurückgesetzt. Auf diese Weise wird verhindert, dass einmal mit SECOS vorgenommene Einstellungen den Betrieb ohne SECOS behindern, ohne dass sie zurückgenommen werden können.

Wird bei einer Rekonstruktion in einem System, in dem das Subsystem SECOS nicht verfügbar ist, eine .BACKUP-Datei verwendet (also RECONUC=B, T oder A), die in einem System gesichert wurde, in dem das Produkt SECOS verfügbar war, so werden alle Privilegien auf die Standard-Einstellungen wie bei First-Start zurückgesetzt. Sollte in diesem Fall die .BACKUP-Datei eine Privilegienverteilung enthalten, die nur in einer höheren BS2000-Version bekannt ist, (dies ist nach Versionwechseln möglich), werden dabei auch die in der aktuellen Version unbekannten Privilegien zurückgesetzt.

Ist sowohl auf dem System des gesicherten als auch des rekonstruierten Benutzerkataloges SECOS im Einsatz, können durch die verstrichene Zeit zwischen Sicherung und Rekonstruktion Kennwort-Verfallsdaten überschritten worden sein. Die drohende Sperre der Benutzerkennung wird verhindert, indem die zum Zeitpunkt der Sicherung vorhandene Restlaufzeit rekonstruiert wird.

### SM-Pubset-Attribute

Wurde der Benutzerkatalog eines SF- bzw. SM-Pubsets gesichert und auf einem SM- bzw. SF-Pubset rekonstruiert, so erhalten die SM-Pubset-spezifischen Attribute den Standardwert (bei einer Umwandlung von SF- nach SM-Pubset) bzw. sie werden gelöscht (bei einer Umwandlung von SM- nach SF-Pubsets).

### Logging-Datei

Anhand der Logging-Datei kann der Systembetreuer das Ergebnis der Rekonstruktion überprüfen. Diese Datei wird unter der Kennung $TSOS angelegt und heißt SYS.SRPM.RECON.LOG.<datum.uhrzeit>. Sie enthält folgende Informationen:

- Pubsets des gesicherten (BACKUP) bzw. zu rekonstruierenden (SYSSRPM) Benutzer-kataloges:

| | |
|---|---|
| PUBSET | Katalogkennung |
| TYPE | SM- oder SF-Pubset |
| DATE | Datum der Sicherung bzw. Rekonstruktion |
| TIME | Uhrzeit der Sicherung bzw. Rekonstruktion |
| ENCRYPT | NO/YES: Logon-Kennwort Verschlüsselung |
| SECOS | NO/YES: Produkt SECOS im Einsatz |

- Benutzerkennung:

| | |
|---|---|
| USERID IN SYSSRPM | |
| USERID | Alle bearbeiteten Benutzerkennungen |
| PRESENT | NO/YES: Eintrag im rekonstruierten Benutzerkatalog |
| USERID IN BACKUP | |
| PRESENT | NO/YES: Eintrag in Backup-Datei |
| ACTION | ADDED/REMOVED: Neuaufnahme/Löschung |
| USERID IN TSOSCAT | |
| PRESENT | NO/YES: Eintrag in TSOSCAT |
| ACTION | ADDED/REMOVED: Neuaufnahme/Löschung |
| DELETED | |
| FILES | Anzahl gelöschter Dateien bei SCOPE=BACKUP |
| JV | Anzahl gelöschter JVs bei SCOPE=BACKUP |

Abhängig vom eingegebenen Kommando IMPORT-PUBSET PUBSET=..., RECONSTRUCT-USERCAT=*BY-BACKUP(SCOPE=*ALL/*BACKUP/*TSOSCAT) werden die Benutzerkennungen bei der Rekonstruktion unterschiedlich behandelt.

*Beispiel*

| USERID IN SYSSRPM | | USERID IN BACKUP | | USERID IN TSOSCAT | | DELETED | |
|---|---|---|---|---|---|---|---|
| USERID | PRESENT | PRESENT | ACTION | PRESENT | ACTION | FILES | JV |
| USERID1 | YES | YES | | YES | | | |
| USERID2 | YES | YES | | NO | ADDED | | |
| USERID3 | YES | NO | ADDED | YES | | | |
| USERID4 | NO | YES | REMOVED | NO | | | |
| USERID5 | NO | NO | | YES | REMOVED | 4 | 3 |

– SCOPE=*ALL/*BACKUP/*TSOSCAT

  USERID1 wurde in der .BACKUP-Datei gesichert und hat bei der Rekonstruktion einen Eintrag im Dateikatalog TSOSCAT.
  USERID1 wird im Benutzerkatalog SYSSRPM mit den gesicherten Benutzerattributen rekonstruiert und behält ihren Dateibestand.

– SCOPE=*ALL/*BACKUP

  USERID2 wurde in der .BACKUP-Datei gesichert, hat aber bei der Rekonstruktion keinen Eintrag im Dateikatalog TSOSCAT.
  USERID2 wird im Benutzerkatalog SYSSRPM mit den gesicherten Benutzerattributen rekonstruiert und erhält einen leeren Eintrag im Dateikatalog.

– SCOPE=*ALL/*TSOSCAT

  USERID3 wurde nicht in der .BACKUP-Datei gesichert, hat aber bei der Rekonstruktion einen Eintrag im Dateikatalog TSOSCAT.
  USERID3 wird im Benutzerkatalog SYSSRPM mit Standardattributen neu eingerichtet und erhält ihren Dateibestand.

– SCOPE=*TSOSCAT

  USERID4 wurde in der .BACKUP-Datei gesichert, hat aber bei der Rekonstruktion keinen Eintrag im Dateikatalog TSOSCAT.
  USERID4 wird nicht in den Benutzerkatalog SYSSRPM übernommen.

– SCOPE=*BACKUP

  USERID5 wurde nicht in der .BACKUP-Datei gesichert, hat aber bei der Rekonstruktion einen Eintrag im Dateikatalog TSOSCAT.
  USERID5 wird nicht in den Benutzerkatalog SYSSRPM übernommen, ihr Eintrag im Dateikatalog wird zusammen mit ihren Dateien und JV's gelöscht.

- Zusammenfassung:

  LOGON PASSWORDS ENCRYPTED: NO/YES:
  Die Logon-Kennwörter wurden verschlüsselt

  EXPIRATION DATES UPDATED: NO/YES:
  Die Logon-Kennwort Verfallsdaten wurden angepasst

  SECOS ATTRIBUTES RESET: NO/YES:
  Die SECOS-Attribute wurden auf Standardwerte gesetzt

  USERIDS RECONSTRUCTED:
  Anzahl der Benutzerkennungen im rekonstruierten Benutzerkatalog

  USERIDS ADDED TO BACKUP:
  Anzahl der Benutzerkennungen, die im neuen Benutzerkatalog gegenüber dem Stand der .BACKUP-Datei hinzugefügt wurden; SCOPE=*ALL/*TSOSCAT

  USERIDS REMOVED FROM BACKUP:
  Anzahl der Benutzerkennungen, die im neuen Benutzerkatalog gegenüber dem Stand der .BACKUP-Datei gelöscht wurden; SCOPE=*TSOSCAT

  USERIDS ADDED TO TSOSCAT:
  Anzahl der Benutzerkennungen in TSOSCAT neu eingerichtet; SCOPE=*ALL/*BACKUP

  USERIDS REMOVED FROM TSOSCAT:
  Anzahl der Benutzerkennungen in TSOSCAT gelöscht; SCOPE=*BACKUP

### Verhalten im Fehlerfall

Im Falle eines Systemfehlers, der zum Abbruch der Rekonstruktion und damit der Import-Verarbeitung führt, hängt die zukünftige Benutzerstruktur vom Stand der Rekonstruktion beim Abbruch ab. Dieser Stand lässt sich anhand von Konsolmeldungen bestimmen:

- vor Meldung `SRM2017`
  Die Rekonstruktion hat noch nicht begonnen, es kann ein erneuter Pubset-Import mit oder ohne Rekonstruktion gestartet werden.

- zwischen Meldung `SRM2017` und `SRM2019`
  Die Rekonstruktion ist in vollem Gange, es kann nur ein erneuter Pubset-Import ohne Rekonstruktion gestartet werden. Die Backup-Datei muss anschließend ggf. wieder eingelesen werden. Die Benutzerstruktur ist auf dem alten Stand.

- nach Meldung `SRM2019`
  Die Rekonstruktion ist so weit fortgeschritten, dass ein Rückzug nicht mehr möglich ist. Es kann ein erneuter Pubset-Import ohne Rekonstruktion gestartet werden. Die Benutzerstruktur ist auf dem neuen Stand.

## 6.4 Benutzerkatalog und SMS-Konzept

Es gibt SF-Pubsets und SM-Pubsets. Die Benutzerverwaltung stellt dafür benutzerspezifischer Kontingente für die von SMS verwalteten verschiedenen Speicherarten bereit.

Alles bisher zum Import von Pubsets und der Sicherung von Benutzerkatalogen gesagte gilt sowohl für SF- wie SM-Pubsets. Die Benutzerkataloge von SF- und SM-Pubsets unterscheiden sich im Prinzip nicht. Deshalb wird im Folgenden auch nur zwischen den beiden Pubset-Typen begrifflich unterschieden, wenn dies notwendig ist.

Um das parallele Betreiben von SF-Pubsets und SM-Pubsets und den Übergang von SF-Pubsets zu SM-Pubsets zu erleichtern, sind die bestehenden Kommandos der Benutzerverwaltung (ADD-USER, MODIFY-USER-ATTRIBUTES, SHOW-USER-ATTRIBUTES usw.) gleichermaßen auf SF- wie SM-Pubsets anwendbar. Bei den für SF-Pubsets bisher relevanten Kontingenten TEMP-SPACE-LIMIT und PUBLIC-SPACE-LIMIT muss jedoch beachtet werden, dass sie bei Anwendung auf SM-Pubsets wie folgt interpretiert werden:

| **SF-Pubset-Attribut** | | **SM-Pubset-Attribut** |
|---|---|---|
| TEMP-SPACE-LIMIT | wird zu | TEMP-SPACE-LIMITS/TOTAL-SPACE |
| PUBLIC-SPACE-LIMIT | wird zu | PERM-SPACE-LIMITS/S0-LEVEL-SPACE |

In SM-Pubsets werden unterschiedliche Speichersysteme (z.B. mit unterschiedlicher Zugriffsgeschwindigkeit) zusammengefasst und gemeinsam verwaltet. Die Speicherarten sind zum Teil von verschiedener Wertigkeit (wie z.B. bei der Zugriffsgeschwindigkeit). Der Umfang der verschiedenen Speicher, über den ein Benutzer verfügen kann, wird daher in einer Hierarchie festgelegt. Die niederwertigeren Speicher umfassen jeweils alle höherwertigen. D.h. die Kontigente an Speicher, die einem Benutzer eingeräumt werden, sind so festzulegen, dass das Kontigent an niederwertigerem Speicher immer größer oder gleich aller höherwertigen Kontingente ist (Kontingenthierarchie).

Die zusätzlichen Kontingente für SM-Pubsets sind nur über das Kommando MODIFY-USER-PUBSET-ATTRIBUTES möglich. Die Kommandos ADD-USER, MODIFY-USER-ATTRIBUTES konnten nicht in kompatibler Weise dafür erweitert werden.
Auch das Kommando MODIFY-USER-PUBSET-ATTRIBUTES ist auf beide Pubset-Typen anwendbar. Es ergänzt allerdings die bestehenden Kommandos zur Spezifizierung von Benutzerattributen nur und ersetzt diese nicht. Bei Anwendung auf ein SF-Pubset werden Angaben zu den zusätzlichen Kontingenten für SM-Pubsets ignoriert. Die neuen Kontingente sind über den Parameter INFORMATION=PUBSET-ATTRIBUTES des Kommandos SHOW-USER-ATTRIBUTES abfragbar.

## 6.5 Bulletin-Datei (Logon-Informationsdatei)

Die Systembetreuung kann eine Informationsdatei einrichten, die unter dem Namen $TSOS.BULLETIN abgespeichert und automatisch an alle Benutzer nach dem Logon ausgegeben wird.

### Aufbau und Inhalt der Bulletin-Datei

Der Name BULLETIN ist für die Logon-Informationsdatei reserviert.

Neben Nachrichten, die an alle Benutzer adressiert sind, können spezielle Informationen an bestimmte Zielgruppen oder einzelne Benutzer gesendet werden. Dem Dialogteilnehmer wird dies durch Ausgabe der folgenden Meldung angeboten:
`CONTINUE userid (Y,N)?`

Bei Eingabe von `N` wird die Ausgabe der Bulletin-Datei abgebrochen.

Die Bulletin-Datei ist so aufzubauen, dass sich der Dialogteilnehmer nach der ersten aktuellen Nachricht wahlweise noch weitere Nachrichten (Folgemitteilungen) ausgeben lassen kann. Dazu ist die folgende Abfrage vor jeder Folgemitteilung einzubauen:
`CONTINUE (Y,N)?` bzw. `CONTINUE userid (Y,N)?`



*Hinweis zum Batch-Betrieb*
Nach SYSLST werden alle benutzerspezifischen Nachrichten ausgegeben.
Nach SYSOUT wird nur die erste Nachricht der Bulletin-Datei ausgegeben.
Die aktuellsten Nachrichten sollten deshalb immer am Anfang der Bulletin-Datei stehen.

Es empfiehlt sich, die Bulletin-Datei wie folgt aufzubauen:

1. Allgemeine Informationen für alle Benutzer
2. Spezielle Informationen für bestimmte Zielgruppen oder einzelne Benutzer
3. Abfragbare Zusatzinformationen für alle Benutzer

### Beispiel

```
/START-EXECUTABLE-PROGRAM FROM-FILE=EDT

  **********************************************************************
  ** Willkommen am Server X/YZ !                                      **
  ** Normaler OPEN SESSION-Betrieb taeglich von 07:00 bis 21:00 Uhr.  **
  **                                                                  **
  ** Service-Nummern:  Bandansage       Tel. 12345                    **
  **                   Listenausgabe    Tel. 34512                    **
  **                   Leitstelle       Tel. 51234                    **
  **********************************************************************
  ** ACHTUNG !                                                        **
  ** Am Freitag wird der Server wegen Wartungsarbeiten                **
  ** bereits um 17:00 runtergefahren.                                 **
  **********************************************************************
**STOP**$EINERLEI
  ** ACHTUNG !  DRINGEND !                                            **
  ** Geben Sie die von Ihnen reservierten Datentraeger MEINER1 und    **
  ** MEINER4 frei. Fuer Rueckfragen: Leitstelle, Tel. 89067           **
  **********************************************************************
**STOP**$ALLERLEI
  ** ACHTUNG !                                                        **
  ** Ihre Privatplatte PRIV03 steht momentan nicht zur Verfuegung.    **
  ** Bitte setzen Sie sich mit uns in Verbindung (Tel. 51234).        **
  **********************************************************************
**STOP**$EINERLEI
  ** Ihre Abrechnungsnummer M0815 wird ungueltig.                     **
  ** Zur Vergabe neuer Abrechnungsnummern: Frau Acco, Tel. 67890      **
  **********************************************************************
**STOP**
  ** In der naechsten Woche wird das Druckerzentrum                   **
  ** umgebaut. In dieser Zeit ist mit Verzoegerungen zu rechnen.      **
  **********************************************************************

W'BULLETIN'
HALT
```

$TSOS.BULLETIN muss mehrbenutzbar sein. Die Datei sollte mit einem Schreib-Kennwort versehen sein oder nur Lesezugriff erlauben:

```
/MODIFY-FILE-ATTRIBUTES BULLETIN,USER-ACCESS=*ALL-USERS,ACCESS=*READ
```

*Ausgabe der Bulletin-Datei*

Der Benutzer mit der Benutzerkennung EINERLEI erhält nach erfolgreichem Logon folgende Ausgabe:

```
% JMS0160 INSTALLATION ' S180-60', BS2000 VERSION 'V180', HOST 'DOYZZEOX': PLE
ASE ENTER '/SET-LOGON-PARAMETERS' OR '?'
/.test logon einerlei,m0815,'denkste#'
% JMS0066 JOB 'TEST' ACCEPTED ON 29-05-12 AT 10:56, TSN = 12CF
 ***************************************************************************
 ** Willkommen am Server X/YZ !                                          **
 ** Normaler OPEN SESSION-Betrieb taeglich von 07:00 bis 21:00 Uhr.     **
 **                                                                      **
 ** Service-Nummern:  Bandansage        Tel. 12345                       **
 **                   Listenausgabe     Tel. 34512                       **
 **                   Leitstelle        Tel. 51234                       **
 ***************************************************************************
 ** ACHTUNG !                                                            **
 ** Am Freitag wird der Server wegen Wartungsarbeiten                    **
 ** bereits um 17:00 runtergefahren.                                     **
 ***************************************************************************
CONTINUE EINERLEI (Y,N)? y
 ** Achtung !  DRINGEND !                                                **
 ** Geben Sie die von Ihnen reservierten Datentraeger MEINER1 und        **
 ** MEINER4 frei. Fuer Rueckfragen: Leitstelle, Tel. 89067               **
 ***************************************************************************
CONTINUE EINERLEI (Y,N)? y
 ** Ihre Abrechnungsnummer M0815 wird ungueltig.                         **
%PLEASE ACKNOwWLEDGE
LTG                                                    TAST
```

```
 ** Zur Vergabe neuer Abrechnungsnummern: Frau Acco, Tel. 67890         **
 ***************************************************************************
CONTINUE (Y,N)? y
 ** In der naechsten Woche wird das Druckerzentrum                       **
 ** umgebaut. In dieser Zeit ist mit Verzoegerungen zu rechnen.         **
 ***************************************************************************
/

:
:

LTG                                                    TAST
```

# 7 POSIX-Benutzerverwaltung

Dieses Kapitel beschreibt die Schnittstellen zur Verwaltung der POSIX-Benutzerattribute einer BS2000-Benutzerkennung. Diese Schnittstellen sind Bestandteil des Bausteins SRPM, der im Software-Produkt SECOS und in BS2000/OSD-BC implementiert ist. Das Software-Produkt SECOS muss jedoch nicht installiert sein, um mit POSIX arbeiten zu können.

Jeder BS2000-Benutzer ist gleichzeitig auch POSIX-Benutzer. Außer einer BS2000-Benutzerkennung mit gültigen individuellen POSIX-Benutzerattributen sind keine weiteren Bedingungen zu erfüllen, um Zugang zu POSIX und seinen Schnittstellen zu erhalten.

Näheres zu SRPM finden Sie im Abschnitt „Beschreibung der Privilegien“ auf Seite 516 und im SECOS-Handbuch „Zugangs- und Zugriffskontrolle“ [48].

Die POSIX-Funktionalität in BS2000/OSD wird ausführlich in den Handbüchern „POSIX-Kommandos“ [40] und „POSIX-Grundlagen“ [41] beschrieben.

#### Was ist POSIX ?

Unter POSIX (Portable Open System Interface for UNIX) versteht man eine Reihe von Standards auf UNIX-Basis. Diese Standards gewährleisten die Kompatibilität und Interoperabilität von Anwendungen in einem heterogenen Netzwerk. Ein heterogenes Netzwerk besteht aus Servern und Produkten von verschiedenen Herstellern sowie aus System- und Benutzersoftware von verschiedenen Software-Anbietern.

Der POSIX-Standard wurde vom Institute of Electrical and Electronics Engineers (IEEE) 1989 als nationaler amerikanischer Standard definiert. Anschließend wurde er vom X/OPEN-Konsortium und 1990 als internationaler Standard verabschiedet (X/OPEN Portability Guide IV, XPG4).

Dem Benutzer stehen die Bibliotheksfunktionen des XPG4-Standards über eine C-Bibliothek und eine definierte Menge von Kommandos über eine Shell (POSIX-Shell) zur Verfügung.

Mit POSIX lassen sich Anwendungsprogramme leicht portieren – unabhängig vom ausführenden Betriebssystem. Deshalb können XPG4-konforme Programme nach einer Neuübersetzung auch in BS2000/OSD ablaufen.
Die POSIX-Programmschnittstellen werden parallel zu den BS2000-Schnittstellen angeboten. Die gemischte Nutzung von BS2000- und POSIX-Programmschnittstellen in einem Programm ist möglich.

Einige Software-Komponenten und -Produkte von BS2000/OSD wurden funktionell erweitert, um POSIX-Dateien bearbeiten zu können. Zum Beispiel können mit SPOOL auch POSIX-Dateien ausgedruckt, sowie mit HSMS POSIX-Dateien, -Dateiverzeichnisse und -Dateisysteme gesichert und rekonstruiert werden.

| **Kommando** | **Bedeutung** |
|---|---|
| ADD-POSIX-USER | Festlegen der POSIX-Benutzerattribute |
| ADD-USER | Erstellen eines Benutzereintrags im Benutzerkatalog [1] |
| MODIFY-LOGON-PROTECTION | Ändern von Schutzattributen [2] |
| MODIFY-POSIX-USER-ATTRIBUTES | Ändern der POSIX-Benutzerattribute einer BS2000-Benutzerkennung |
| MODIFY-POSIX-USER-DEFAULTS | Ändern der POSIX-Standardattribute eines Pubsets |
| MODIFY-USER-ATTRIBUTES | Ändern des Katalogeintrags eines Benutzers [1] |
| SET-LOGON-PROTECTION | Vereinbaren von Schutzattribute [2] |
| SHOW-LOGON-PROTECTION | Anzeigen von Schutzattributen [2] |
| SHOW-POSIX-STATUS | Anzeigen des POSIX-Status |
| SHOW-POSIX-USER-ATTRIBUTES | Anzeigen der POSIX-Benutzerattribute einer BS2000-Benutzerkennung |
| SHOW-POSIX-USER-DEFAULTS | Anzeigen der POSIX-Standardattribute eines Pubsets |
| SHOW-USER-ATTRIBUTES | Ausgabe von Informationen über die Einträge im Benutzerkatalog [1] |
| START-POSIX-SHELL | POSIX-Shell zur Verfügung stellen |
| **Makro** | **Bedeutung** |
| SRMUINF | Lesen der Daten aus dem Benutzerkatalog und Übertragen in einen vorher festgelegten Bereich |

Tabelle 17: BS2000-Schnittstellen zur POSIX-Benutzerverwaltung

[1] Kommandos zum Verwalten der Abrechnungsnummer für den Zugang über einen fernen Server.

[2] Kommandos zum Verwalten der Zugangsberechtigung über einen fernen Server:
Wenn das Software-Produkt SECOS eingesetzt wird, kann für bereits existierende BS2000-Benutzerkennungen festgelegt werden, ob sich der Benutzer eines fernen Servers mit dem UNIX-Kommando rlogin Zugang zum System (POSIX) verschaffen darf. Dazu steht der Operand POSIX-RLOGIN-ACCESS=*YES(PASSWORD-CHECK=*YES/*NO) bzw. POSIX-RLOGIN-ACCESS=*NO bei den Kommandos SET- und MODIFY-LOGON-PROTECTION zur Verfügung.

Diese und weitere POSIX-Kommandos sind in den Handbüchern „SECOS - Zugangs- und Zugriffskontrolle““ [48] und „Kommandos“ [27] beschrieben.

### POSIX-Benutzerattribute

Siehe das Kapitel „POSIX-Benutzer verwalten“ im Handbuch „POSIX-Grundlagen“ [41].

### Privilegierung zur Verwaltung der POSIX-Benutzerattribute

Für POSIX gibt es das Privileg POSIX-ADMINISTRATION. Inhaber dieses Privilegs werden POSIX-Verwalter genannt. Sie haben folgende Aufgaben und Rechte:

- Verwaltung der POSIX-Benutzerattribute aller BS2000-Benutzerkennungen auf allen Pubsets
- Verwaltung der POSIX-Standardattribute auf allen Pubsets
- Aufruf von privilegierten POSIX-Funktionen

Das Privileg POSIX-ADMINISTRATION ist standardmäßig an die Systemkennung SYSROOT geknüpft. Dieses Privileg kann SYSROOT nicht entzogen werden.
Der Sicherheitsbeauftragte (Privileg SECURITY-ADMINISTRATION) kann dieses Privileg auch anderen BS2000-Benutzerkennungen verleihen und entziehen.

SYSROOT ist das POSIX-Gegenstück zur Systemverwalterkennung *root* in UNIX-Systemen. SYSROOT wird beim First-Start des BS2000-Systems eingerichtet und erhält automatisch die Benutzernummer 0. SYSROOT kann keine andere Benutzernummer zugewiesen werden.

Inhaber des Privilegs USER-ADMINISTRATION erhalten zusätzlich die Berechtigung, die POSIX-Benutzerattribute und die Standardwerte für die POSIX-Benutzerattribute zu verwalten. Sie sind diesbezüglich dem POSIX-Verwalter gleichgestellt.

Die Berechtigung des Gruppenverwalters der Gruppe *UNIVERSAL wird auf die POSIX-Benutzerattribute ausgedehnt. Er ist bei der Verwaltung der POSIX-Benutzerattribute auf dem von ihm verwalteten Pubset den Inhabern des Privilegs USER-ADMINISTRATION gleichgestellt. Deshalb gelten für ihn nicht die im Folgenden aufgeführten Einschränkungen für Gruppenverwalter seiner Hierarchie.

Gruppenverwalter dürfen ebenfalls die POSIX-Benutzerattribute verwalten. Allerdings gelten für sie folgende Einschränkungen:

- Sie können nicht die POSIX-Standardattribute verwalten.
- Die Art der POSIX-Benutzerattribute, die ihrer Verwaltung unterstellt sind, hängt von ihrer Autorisierung ab (ADM-AUTHORITY).
- Der Wertebereich der POSIX-Benutzerattribute ist für sie eingeschränkt.
- Sie können nur die Gruppen- und Untergruppenmitglieder verwalten, die ihnen unterstellt sind.

Für weitere Informationen zum Thema Privilegierung siehe Abschnitt „Privilegien“ auf Seite 511.

#### Benutzernummer einer BS2000-Benutzerkennung zuordnen

Siehe das Kapitel „POSIX-Benutzer verwalten“ im Handbuch „POSIX-Grundlagen“ [41].

#### BS2000- und POSIX-Gruppen verwalten

Siehe das Kapitel „POSIX-Benutzer verwalten“ im Handbuch „POSIX-Grundlagen“ [41].

#### Neue POSIX-Benutzer eintragen

Siehe das Kapitel „POSIX-Benutzer verwalten“ im Handbuch „POSIX-Grundlagen“ [41].

#### POSIX-Benutzerattribute im POSIX-Dateisystem abbilden

Das POSIX-Benutzerattribut „Benutzernummer“ ist eng mit dem POSIX-Dateisystem verbunden: Die Benutzernummer dokumentiert, wer der Eigentümer einer Datei ist. Anders als in BS2000/OSD kann der Root-Berechtigte einer Datei oder einem Dateiverzeichnis auf einfache Weise einen neuen Eigentümer zuweisen (POSIX-Kommando *chown*).

#### Benutzerinformationen per Programm lesen

Siehe das Kapitel „POSIX-Benutzer verwalten“ im Handbuch „POSIX-Grundlagen“ [41].

# 8 Dateiverwaltung

## 8.1 Dateikatalog

Der Dateikatalog eines SF- oder eines SM-Pubsets dient als Behälter für die Katalogeinträge der dem Pubset zugehörigen Dateien auf gemeinschaftlichen Plattenspeichern und Net-Storage, den Jobvariablen, sowie auch für die Katalogeinträge von Privatplattendateien und Banddateien.

Sämtliche Benutzer- und Systemdateien sowie Jobvariablen haben einen Eintrag im Dateikatalog. Die Dateien und Jobvariablen werden über einen eindeutigen Namen identifiziert.

Die **Namensstruktur** besteht aus:

*:catid:$userid.name*

catid Der jeweilige Pubset wird über die Katalogkennung (catid) adressiert.
Sie ist max. vier Zeichen lang.
(siehe auch Abschnitt „VSN und Pubset-Adressierung“ auf Seite 316).

userid Die Benutzerkennung (userid) ist maximal 8 Zeichen lang.

name Der Name für die Datei bzw. Jobvariable darf maximal 41 Zeichen lang sein.
Bei Verwendung einer mehrstelligen Katalogkennung und maximaler Länge der Benutzerkennung reduziert sich diese Zahl entsprechend.

catid, userid und name dürfen insgesamt nicht länger als 54 Zeichen sein.

| Kommando | Bedeutung |
|---|---|
| CHECK-IMPORT-DISK-FILE | Importieren von Dateien / Dateigenerationen überprüfen |
| CREATE-FILE | Namen und Merkmale einer neuen Datei vereinbaren |
| CREATE-FILE-GENERATION | Dateigeneration einer Dateigenerationsgruppe erzeugen |
| CREATE-FILE-GROUP | Dateigenerationsgruppe erzeugen |
| DELETE-FILE | Datei löschen |
| DELETE-FILE-GENERATION | Dateigeneration einer Dateigenerationsgruppe löschen |
| DELETE-FILE-GROUP | Dateigenerationsgruppe mit zugehörigen Dateigenerationen löschen |
| EXPORT-FILE | Katalogeintrag für Dateien auf privaten Datenträgern oder Net-Storage-Volumes löschen |
| IMPORT-FILE | Katalogeintrag für Dateien auf privaten Datenträgern oder Net-Storage-Volumes erstellen |
| MODIFY-FILE-ATTRIBUTES | Merkmale einer Datei ändern |
| MODIFY-FILE-GENERATION-SUPPORT | Merkmale einer Dateigeneration ändern |
| MODIFY-FILE-GROUP-ATTRIBUTES | Merkmale einer Dateigenerationsgruppe ändern |
| REMOVE-CE-LOCK | Sperre des Katalogeintrags zurücksetzten |
| REPAIR-FILE-LOCKS | unberechtigte Dateisperre zurücksetzen |
| SHOW-CE-LOCK | Sperre des Katalogeintrags anzeigen |
| SHOW-FILE-ATTRIBUTES | Merkmale einer Datei ermitteln |
| SHOW-FILE-LOCKS | Dateisperren anzeigen |
| SHOW-PUBSET-CATALOG-ALLOCATION | Informationen über Kataloge eines Pubsets ausgeben |

Tabelle 18: Kommandoübersicht zur Verwaltung des Dateikataloges und der Katalogeinträge

## Aufbau eines Dateikataloges

Der Dateikatalog enthält für jede auf dem Pubset existierende Datei, also auch für sich selbst, einen Eintrag, in dem u.a. die Dateimerkmale, die Schutzkriterien und die Lage der Datei auf dem Datenträger vermerkt sind.

Im ersten Katalogblock (4K-Block) sind Daten zur Verwaltung des Dateikatalogs hinterlegt. Die Katalogeinträge (Catalog Entries; CEs) stehen in den nachfolgenden Katalogblöcken. Der Katalog ist nach Benutzerkennungen strukturiert. Ein Block eines Katalogs enthält jeweils nur CEs einer Benutzerkennung; die CEs sind unsortiert sequenziell abgespeichert. Reicht der Platz im Block zur Aufnahme zusätzlicher CEs nicht mehr aus, wird ein weiterer Block angefordert und mit dem letzten Block dieser Benutzerkennung verkettet.
Alle Benutzer im Katalog werden zusammen mit dem Beginn ihrer Benutzerkette (PBN, Primary Block Number) in der PBN-Tabelle des Katalogs verwaltet.
Bei Aufnahme einer neuen Benutzerkennung in den Benutzerkatalog eines SF-Pubsets wird automatisch ein neuer Katalogblock reserviert, der die PBN dieser Kennung ist. Die logische Blocknummer (LBN) dieses Katalogblocks wird zusammen mit der Benutzerkennung in der PBN-Tabelle eingetragen.
In einem SM-Pubset wird für den neuen Benutzer zunächst nur ein Eintrag in der PBN-Tabelle des Control-Volume-Set-Katalogs erzeugt. Die PBN-Vergabe in einem SM-Pubset-Katalog erfolgt erst dann, wenn in diesem Katalog der erste CE des Benutzers abgelegt wird.

Der zweite 4K-Block des Dateikatalogs enthält immer den CE des Dateikatalogs selbst und ist der Beginn der Benutzerkette der Kennung TSOS.

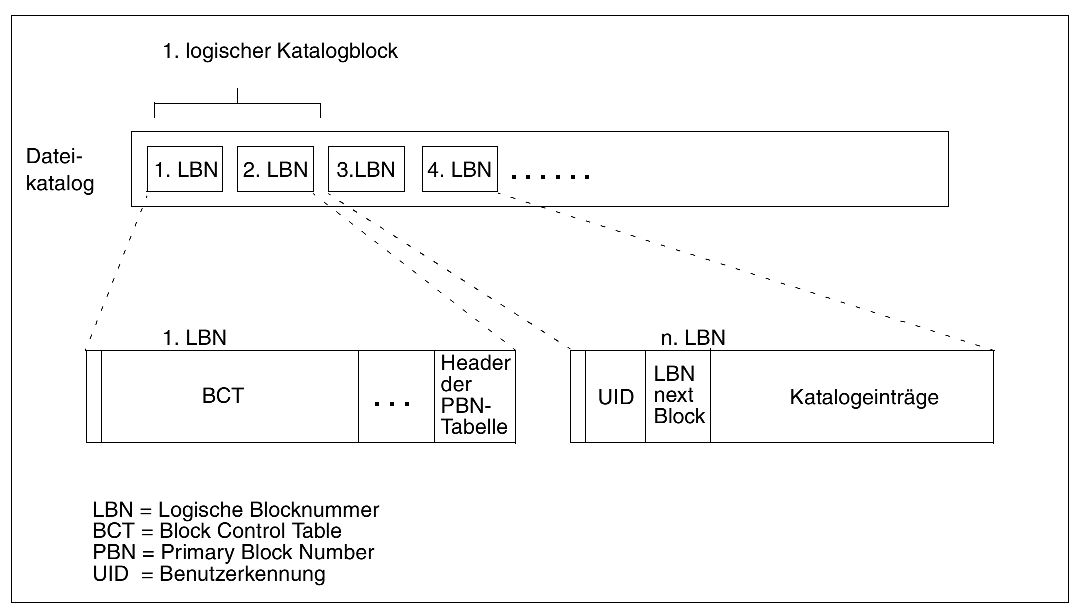



Bild 13: Aufbau des Dateikatalogs

Die logische Blockgröße im TSOSCAT beträgt 4KB, die logische Blockummer (LBN) wird als Nummer der logisch kleinsten, im 4-KB-Block enthaltenen PAM-Seite geführt.

Die maximale Größe des Dateikataloges TSOSCAT ist abhängig vom TSOSCAT-Typ (siehe Seite 285) und beträgt:

- für TSOSCAT-Typ EXTRA-LARGE 32.008 4-KB-Blöcke oder 64.016 PAM-Seiten
- für TSOSCAT-Typ LARGE 16.184 4-KB-Blöcke oder 32.368 PAM-Seiten
- für TSOSCAT-Typ NORMAL 8.192 4-KB-Blöcke oder 16.384 PAM-Seiten

Die Größe des Dateikatalogs legt die Systembetreuung fest in Abhängigkeit von:

- Anzahl der Benutzerkennungen
- Anzahl der Dateien pro Benutzerkennung
- Anzahl der Jobvariablen pro Benutzerkennung

Ein Katalogblock kann 1 bis 13 Katalogeinträge aufnehmen. Die Maximallänge eines Eintrags beträgt 4079 Byte, die Minimallänge (mit 1 Byte langem Dateinamen) 301 Byte, solange kein Platz zugewiesen ist. Ansonsten erhöht sich dieser Wert um die Länge der Extent-Liste. Diese beträgt mindestens 14 Byte bei Volumes < 32 GB und 16 Byte bei Volumes ≥ 32 GB.

Von der Struktur her besteht kein Unterschied zwischen den Katalogen eines System-Managed-Pubsets (SM-Pubset) und dem Katalog eines Single-Feature-Pubsets (SF-Pubset). Die Realisierung des Dateikatalogs ist jedoch unterschiedlich.

### Dateikatalog auf SF-Pubsets

Der Dateikatalog ist in der Datei mit dem Namen :<catid>:$TSOS.TSOSCAT untergebracht. Er beinhaltet sämtliche Benutzer- und Systemdateien sowie die Jobvariablen.

### Dateikatalog auf SM-Pubsets

Im Gegensatz zu SF-Pubsets ist der Katalog eines SM-Pubsets aus mehreren Katalogdateien aufgebaut. Der Aufbau wird in folgendem Bild dargestellt.

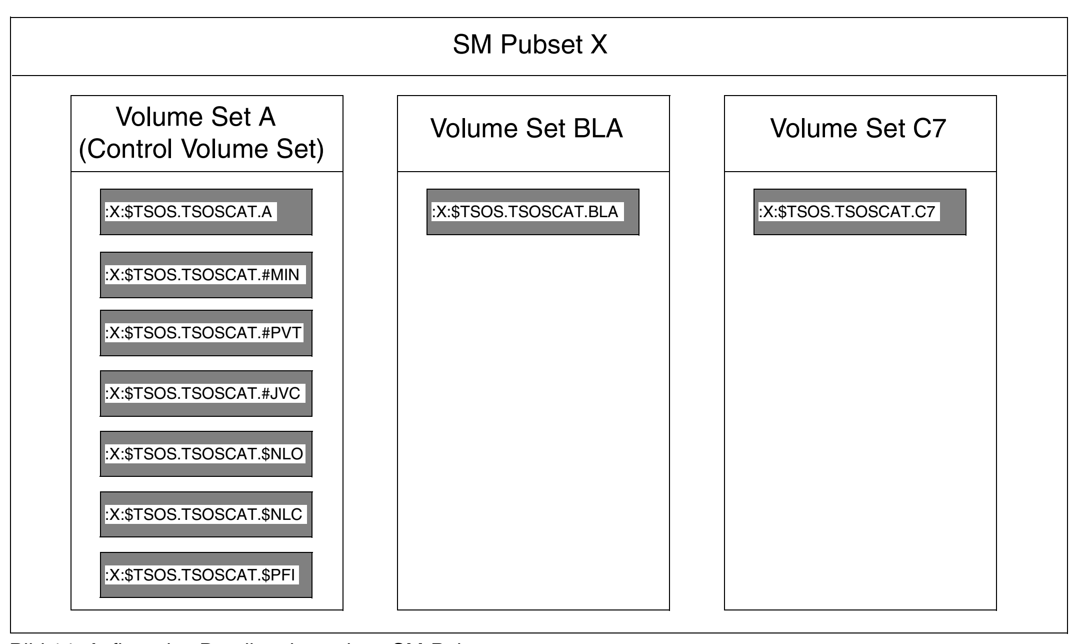



Bild 14: Aufbau des Dateikatalogs eines SM-Pubsets

- Jeder Volume-Set eines SM-Pubsets enthält eine volume-set-spezifische Katalogdatei mit den Katalogeinträgen für die auf ihm liegenden Dateien. Der Pfadname der volume-set-spezifischen Katalogdatei ergibt sich aus Pubset-Kennung und Volume-Set-Kennung wie folgt: :<catid>:$TSOS.TSOSCAT.<volume-set-id>.
- Zusätzlich benötigt das Katalogverwaltungssystem (CMS) weitere Katalogdateien, welche alle auf dem Control-Volume-Set des SM-Pubsets liegen:
  - Katalogeinträge für auf eine Hintergrundebene migrierte Dateien sowie katalogisierte Dateien, die keinen Platz belegen, werden in der Katalogdatei :<catid>:$TSOS.TSOSCAT.#MIN aufbewahrt.
  - Als Behälter für Katalogeinträge von Dateien auf Net-Storage, Banddateien und Privatplattendateien dient die Katalogdatei :<catid>:$TSOS.TSOSCAT.#PVT.
  - Als Behälter für die Jobvariablen dient die Katalogdatei :<catid>:$TSOS.TSOSCAT.#JVC.

Neben den verschiedenen Katalogen benutzt das CMS im SM-Pubset noch weitere Dateien, die sich ebenfalls alle auf dem Control-Volume-Set des SM-Pubsets befinden:

– Die Katalog-Index-Datei :<catid>:$TSOS.TSOSCAT.$PFI dient der Sicherung des Katalog-Index bei der Außerbetriebnahme eines SM-Pubsets. Sie ist eine den SM-Pubset-Katalogen übergeordnete Verwaltungsinstanz, die Informationen über die Lage einer Datei oder einer Jobvariablen im SM-Pubset enthält.

– Die Datei :<catid>:$TSOS.TSOSCAT.$NLO beinhaltet die Namensliste (Name-List) und dient der Datensicherheit. Bei Ausfall eines Volume-Sets können mit ihrer Hilfe die durch den Ausfall verlorengegangenen Daten rekonstruiert werden.
In der Namensliste existiert ein Eintrag für jede Datei, die in einem Volume-Set des SM-Pubsets existiert. Auch temporäre Dateien besitzen einen solchen Verweis. Lediglich für Jobvariablen und Dateien, die in Katalogen auf dem Control-Volume-Set abgelegt sind, existieren keine solchen Verweise, denn mit Ausfall des Control-Volume-Sets ist auch kein Zugriff mehr auf die Namensliste möglich.
Der Eintrag für einen CE in der Namensliste besteht aus der Kennung des Katalogs, in dem der CE abgelegt ist, der Benutzerkennung des Datei-Eigentümers und dem Dateinamen.

– Die Datei :<catid>:$TSOS.TSOSCAT.$NLC beinhaltet die Namensliste-Kopie (Name-List-Copy) und wird zur Rekonstruktion der Namensliste nach einem Systemabsturz benötigt.
Die Namensliste und die Namensliste-Kopie sollten gleich groß angelegt werden, weil bei jeder Rekonstruktion des Katalog-Indexes die Namensliste zur Namensliste-Kopie wird und umgekehrt. Bei der Verwaltung der Namensliste könnte das CMS dann nur die kleinere der beiden Dateigrößen berücksichtigen.

### Anlegen und Erweitern von Katalogdateien

Das Anlegen der Katalogdateien erfolgt implizit beim Einrichten des Pubsets.
Bei SM-Pubsets sind die Volume-Sets, auf denen die einzelnen Katalogdateien liegen, entsprechend der oben gegebenen Beschreibung festgelegt.

Die maximale Größe der Katalogdateien, in welchen Katalogeinträge abgelegt sind, ist:

- 64.016 PAM-Seiten für TSOSCAT-Typ EXTRA-LARGE
- 32.368 PAM-Seiten für TSOSCAT-Typ LARGE
- 16.384 PAM-Seiten für TSOSCAT-Typ NORMAL

Die Größe und Verteilung der bei der Generierung erzeugten Katalogdateien auf bestimmte Volumes sind durch den Systembetreuer beeinflussbar.

Die Katalogdateien können im laufenden Betrieb z.B. mit dem Kommando MODIFY-FILE-ATTRIBUTES ...,SPACE=... erweitert werden. Dies gilt sowohl auf SM- als auch auf SF-Pubsets. Die Vergrößerung des Katalogs wird sofort wirksam, nicht erst nach dem

nächsten Import des Pubsets. Bei 90%iger Auslastung des Katalogs wird dieser u.U. vom System automatisch erweitert (siehe „Automatische Vergrößerung des Katalogs“ auf Seite 286).

Wurde für einen SF-Pubset das Produkt SCA aktiviert, so wird SCA bei einer Erweiterung der Datei TSOSCAT implizit vom System beendet und wieder neu gestartet (zur Anpassung der SCA-Tabellenstruktur). Dies führt i.a. zu keiner Beeinträchtigung des Systems.

### Katalogformate

Es gibt drei Katalogformate:
NORMAL und LARGE (implizit, für Pubsets mit Objekten > 32 GB) sowie EXTRA LARGE.

| TSOSCAT-Typ | max. Größe (PAM-Seiten) | kompatibel zu |
|---|---|---|
| NORMAL | 16.384 | BS2000/OSD-BC V1.0 |
| LARGE | 32.368 | BS2000/OSD-BC V5.0 |
| EXTRA LARGE | 64.016 | BS2000/OSD-BC V6.0B |

Die Katalogformate LARGE und EXTRA LARGE sind inkompatibel: Werden sie für einen Pubset genutzt, so ist mit diesem Pubset ein Rückstieg in eine ältere BS2000-Version nicht mehr möglich.

Pubsets können beim Importieren oder beim Einrichten mit SIR mit TSOSCAT-Typ EXTRA-LARGE konvertiert werden:

- Kommando IMPORT-PUBSET ...,EXTRA-LARGE-CAT-CONV=*YES

  Bei einem SM-Pubset wirkt sich diese Angabe auf alle Kataloge des Pubsets aus. Sind Volume-Sets des SM-Pubsets beim Import im Zustand „in hold“, so werden deren Kataloge erst bei der Aufhebung des „in hold“-Zustands konvertiert.

- SIR-Anweisung DECLARE-PUBSET ..., TSOSCAT-TYPE=*EXTRA-LARGE

Für SM-Pubsets mit Katalogen des Typs EXTRA LARGE können für jeden der Spezialkataloge #MIN, #JVC und #PVT bis zu 100 Teilkataloge mit jeweils 64.016 PAM-Seiten angelegt werden. Dazu erfolgt die Umbenennung der Spezialkataloge beim Konvertieren in den Typ EXTRA LARGE:

- <catid>$TSOS.TSOSCAT.#M00
- <catid>$TSOS.TSOSCAT.#J00
- <catid>$TSOS.TSOSCAT.#P00

Bis zu 99 weitere Teilkataloge können hinzugefügt werden, entweder explizit mit dem Kommando ADD-CATALOG-FILE (siehe Handbuch „Kommandos“ [27]) oder implizit und automatisch durch das Katalogverwaltungssystem CMS (siehe nächsten Abschnitt).

### Automatische Vergrößerung des Katalogs

Das CMS erkennt, wenn ein Katalog zu 90% belegt ist, und nimmt automatisch eine Katalogvergrößerung vor, sofern das ohne Wechsel des Katalogformats möglich ist.

Für Spezialkataloge des Typs EXTRA LARGE wird dann ein neuer Teilkatalog angelegt, wenn keiner der bereits existierenden Teilkataloge vergrößert werden kann.

### Ausgabe der aktuellen Katalogsituation

Das Kommando SHOW-PUBSET-CATALOG-ALLOCATION gibt den Katalogtyp (NORMAL, LARGE, EXTRA LARGE), den Belegungsgrad pro Katalog (Nutzung/Dateigröße) und die Erweiterbarkeit pro Katalog aus.
Diese Ausgabe kann entweder für einen einzelnen Pubset oder für alle an einem Server lokal und im Master-Modus importierten Pubsets erfolgen.

### Dateikatalog auf Net-Storage-Volumes

Jedes Net-Storage-Volume hat einen eigenen Dateikatalog mit dem Namen `.BS2FSCAT`.
Er enthält für alle Dateien, die auf diesem Net-Storage-Volume abgelegt sind, jeweils eine Kopie des Katalogeintrags im TSOSCAT des Pubsets, dem das Net-Storage-Volume zugeordnet ist.

### 8.1.1 Leistungsverbesserung der Katalogverwaltung

- Es ist sinnvoll, die Kennung der Systemverwaltung aufzuteilen in:

  a) Systemverwaltungskennung (TSOS)
  Unter dieser Kennung sind alle für die Aufrechterhaltung des Server-Betriebes notwendigen Dateien katalogisiert. Dabei handelt es sich z.B. um den Benutzerkatalog, den Dateikatalog, Protokolldatei, Abrechnungsdatei, aber auch alle Dateien, auf die schon vor SYSTEM READY zugegriffen werden muss.

  b) Standard-Benutzerkennung
  Diese Kennung wird mit dem Systemparameter DEFLUID festgelegt (siehe Abschnitt „Systemparameter“ auf Seite 732).
  Sämtliche Dienstprogramme, Sprachübersetzer, Bibliotheken usw., die vom Benutzer häufig benutzt werden, sollten unter dieser Kennung katalogisiert sein.
  Bei diesem Parameter kann auch eine Pubset-Angabe gemacht werden. Sinnvoller ist es allerdings, die Standard-Pubset-Angabe über das Kommando ADD-USER zu vergeben.
  Falls der Pubset mit der bei DEFLUID angegebenen Benutzerkennung ausfällt, kann ein Reserve-Pubset benutzt werden. Es muss dann nur der Benutzerkatalog-Eintrag der Benutzerkennung geändert werden.

  Die Aufteilung der Kennung der Systemverwaltung ist aus folgenden Gründen sinnvoll: Die Anzahl der Systemdateien, die unter TSOS stehen müssen, beträgt mehrere Tausend. Ohne Aufteilung, d.h. bei DEFLUID=TSOS, müssen bei jedem Benutzerzugriff auf $datei auch diese Katalogeinträge immer mit durchsucht werden.

- Der Benutzer hat bei den Kommandos START-/LOAD-EXECUTABLE-PROGRAM, CALL-PROCEDURE, ENTER-PROCEDURE und ENTER-JOB die Möglichkeit, entweder *$dateiname* oder nur *dateiname* anzugeben.
  Gibt er z.B. START-EXECUTABLE-PROGRAM FROM-FILE=$EDT wird sofort der Bezug zu der bei DEFLUID angegebenen Benutzerkennung hergestellt.
  Bei Angabe von START-EXECUTABLE-PROGRAM FROM-FILE=EDT wird zuerst unter der aufrufenden (eigenen) Benutzerkennung nach einer Datei mit Namen EDT gesucht und danach unter der bei DEFLUID angegebenen Benutzerkennung.

- Die Sekundärspeicherplatz-Zuweisung ist sehr zeitaufwändig, da bei jeder Anforderung durch die Benutzertask der entsprechende Katalogeintrag geändert wird.
  Die Systemverwaltung sollte deshalb einen geeigneten Wert für die Sekundärspeicherplatz-Zuweisung festlegen (Systemparameter DMSCALL, siehe ab Seite 732).

- Um einer Sättigung des gemeinschaftlichen Speicherplatzes wirksam vorzubeugen, ist es ratsam, unbenutzten Speicherplatz der Benutzerdateien freizugeben.
  Dazu kann die Systemverwaltung Prozeduren erstellen.
  Siehe Abschnitt „Überwachung der Speicherplatz-Sättigung“ auf Seite 393.

## 8.1.2 Zugriffssperren aufheben

**Sperre des Katalogeintrags (CE-Lock)**

Um die Konsistenz von Katalogeinträgen zu gewährleisten (z.B. der Server-Kommunikation) können Systemkomponenten den Katalogeintrag einer Datei bzw. JV zeitweise gegen Änderungen sperren. Solange ein CE-Lock besteht, kann der betroffene Katalogeintrag der Datei bzw. JV nicht geändert werden. Kann ein CE-Lock z.B. wegen eines Fehlers bei der Server-Kommunikation nicht mehr zurückgesetzt werden, bleibt der Katalogeintrag weiter gesperrt, obwohl die Sperre nicht mehr notwendig wäre.

Solch ein „hängender“ CE-Lock behindert die Arbeit von Tasks, die diese Datei oder JV für Verarbeitungsvorgänge benötigen. Er kann nur durch die Systembetreuung beseitigt werden:

Mit dem Kommando REMOVE-CE-LOCK kann die Systembetreuung eine Sperre für den Katalogeintrag einer Datei oder JV zurücksetzen.

Dabei ermittelt das System, ob für den Katalogeintrag der angegebenen Datei oder JV eine Sperre (CE-Lock) existiert. Wenn ein CE-Lock existiert, wird die Task des Lock-Halters ermittelt.
Ist die Task nicht mehr vorhanden oder befindet sie sich im Zustand PENDING INDEFINITELY, handelt es sich um einen „hängenden“ CE-Lock, der z.B. wegen Fehlers bei der Server-Kommunikation nicht mehr zurückgesetzt werden konnte. Der „hängende“ CE-Lock wird zurückgesetzt.
Befindet sich die Task des Lock-Halters in einem anderen Zustand kann der CE-Lock nicht freigegeben werden und das Kommando wird zurückgewiesen. In diesem Fall ist zu prüfen, ob die Task des Lock-Halters beendet werden kann (z.B. Kommando CANCEL-JOB). Mit dem Kommando SHOW-CE-LOCK kann die TID dieser Task und die Sysid des Systems, in dem die Task arbeitet, ermittelt werden.

Für Shared Pubsets können die Kommandos REMOVE- und SHOW-CE-LOCK von einem beliebigen System des Verbunds eingegeben werden.

### Dateisperre aufheben

Mit dem Kommando REPAIR-FILE-LOCKS hebt der Dateieigentümer bzw. die Systembetreuung (Benutzerkennung TSOS) „unberechtigte“ Dateisperren für eine Datei auf. Dabei handelt sich um Dateisperren, die nicht mehr notwendig sind, aber aus folgenden Gründen vom System nicht mehr automatisch zurückgesetzt werden konnten:

- kurzfristiger Verbindungsausfall in einem Rechnerverbund
- Systemfehler verhinderte das Rücksetzen einer Dateisperre

Über die aktuell wirksamen Sperren einer Datei kann sich der Benutzer mit dem Kommando SHOW-FILE-LOCKS informieren.

Die für eine Datei vorhandenen Sperren können ausschließlich einer der nachfolgenden Arten zugeordnet werden:

- Sperren, die durch einen Job gesetzt sind
- Sperren, die durch eine Systemkomponente gesetzt sind
- Sperren, die für die Dateiübertragung gesetzt sind

Das Kommando SHOW-FILE-LOCKS informiert den Dateieigentümer und alle zugriffsberechtigten Benutzer über die aktuell wirksamen Sperren einer Datei. Es dient dem Benutzer zur Diagnose von Verarbeitungsproblemen wegen einer bestehenden Dateisperre.
Angezeigt werden Dateisperren, die folgende Ursachen besitzen:

- Die Datei ist gerade geöffnet.
- Für die Datei wurde eine explizite Reservierung mit dem Kommando SECURE-RESOURCE-ALLOCATION vorgenommen.
- Zum Ausdrucken der Datei wurde eine Sperre vereinbart. Die Sperre kann für eine auszudruckende Datei explizit im Kommando PRINT-DOCUMENT (Operand LOCK-FILE=*YES) bzw. im Makroaufruf PRNT vereinbart werden. In der SPOOL-Parameterdatei kann als Standard-Wert für LOCK-FILE mit YES die Dateisperre vereinbart sein (siehe Kommando SHOW-SPOOL-PARAMETERS).
  Eine Datei, die gerade auf den Drucker ausgegeben wird, ist in jedem Fall bis zur Beendigung der Ausgabe gesperrt.
- Die Datei ist für eine Dateiübertragung reserviert (siehe Kommando TRANSFER-FILE).
- Es handelt sich um eine SYSLST-Datei, die auf das Ausdrucken nach Job-Beendigung wartet.
- Die Datei wird gerade von einem Concurrent-Copy-Auftrag bearbeitet.
- Die Datei ist die Quelldatei für einen Batch-Job, der noch in der Warteschlange steht.

Nicht angezeigt werden Dateisperren, die auf Grund des Katalogeintrags wirksam sind. Diese Sperren werden mit dem Kommando SHOW-CE-LOCK angezeigt.

## 8.2 ACS: Alias-Katalogsystem

Mit ACS (Alias Catalog Service) ist es möglich, auf Dateien und Jobvariablen unter zusätzlichen, in gewissen Grenzen frei wählbaren Namen zuzugreifen. Dem Benutzer wird die Möglichkeit geboten, für die von ihm benötigten Dateien/JVs Aliasnamen zu definieren und zusammen mit der Zuordnung zu realen Dateien/JVs in speziellen Katalogen, den Aliaskatalogen, zu hinterlegen. Bei der Bearbeitung der Datei/JV ist dann nur noch der Aliasname anzugeben. Die Kataloge werden tasklokal geführt.

Detaillierte Informationen zu ACS finden Sie im Handbuch „Einführung in das DVS“ [19].

| **Kommando** | **Bedeutung** |
|---|---|
| ADD-ACS-SYSTEM-FILE | Neue Identifikation für eine AC-Systemdatei vereinbaren und der Identifikation einen Dateinamen zuordnen |
| ADD-ALIAS-CATALOG-ENTRY | Eintrag im Aliaskatalog erzeugen und Wirkungsbereich der Ersetzung festlegen |
| HOLD-ALIAS-SUBSTITUTION | ACS-Funktion Namensersetzung unterbrechen |
| LOAD-ALIAS-CATALOG | Gespeicherte Einträge aus einer AC-Datei in den Aliaskatalog übernehmen |
| MODIFY-ACS-OPTIONS | ACS-Optionen tasklokal oder systemweit verändern |
| MODIFY-ACS-SYSTEM-FILE | Definition einer AC-Systemdatei verändern |
| MODIFY-ALIAS-CATALOG-ENTRY | Bestehenden Eintrag im Aliaskatalog modifizieren |
| PURGE-ALIAS-CATALOG | Aktuellen Aliaskatalog löschen |
| REMOVE-ACS-SYSTEM-FILE | Definition einer AC-Systemdatei löschen |
| REMOVE-ALIAS-CATALOG-ENTRY | Bestehenden Eintrag im Aliaskatalog löschen |
| RESUME-ALIAS-SUBSTITUTION | ACS-Funktion Namensersetzung wiederaufnehmen |
| SET-FILE-NAME-PREFIX | Namenspräfix vereinbaren und Wirkungsbereich der Ersetzung festlegen |
| SHOW-ACS-OPTIONS | Gültige ACS-Optionen anzeigen |
| SHOW-ACS-SYSTEM-FILES | Identifikationen der AC-Systemdateien anzeigen |
| SHOW-ALIAS-CATALOG-ENTRY | Informationen über die Aliaskatalogeinträge anfordern |
| SHOW-FILE-NAME-PREFIX | Aktuellen Präfix und seinen Gültigkeitsbereich anzeigen |
| START-ACS | ACS systemweit aktivieren |
| STORE-ALIAS-CATALOG | Aliaskatalog in Datei abspeichern |

Tabelle 19: Kommandoübersicht für das Alias-Katalogsystem

# 8.3 PFA: Performant File Access

## 8.3.1 Das HIPERFILE-/PFA-Konzept

Unter dem Begriff „HIPERFILE-Konzept“ (High Performance Files) werden in BS2000/OSD verschiedene Erweiterungen im Bereich der Systemsoftware und der Hardware zusammengefasst, die durch „Caching“ von Dateien den Dateizugriff beschleunigen und einen eventuellen I/O-Engpass vermeiden sollen. Als Puffer- (bzw. Cache-)Speicher kommen Halbleiter-Speicher mit kurzen Zugriffszeiten zum Einsatz, die eine Anpassung zwischen der Zugriffszeit zum Hauptspeicher und der (längeren) Zugriffszeit zu Festplattenspeichern ermöglichen.

Das HIPERFILE-Konzept von BS2000/OSD bietet Caching in den unterschiedlichen Cache-Medien sowohl über die Kommando-Oberfläche des jeweils beteiligten Subsystems als auch über eine in das DVS integrierte einheitliche Kommando-Oberfläche:

- ADM-PFA (Administrator Performant File Access):
  Caching in den Medien Hauptspeicher und Globalspeicher (GS) über privilegierte Kommandos des kostenpflichtigen Produkts DAB
- PFA (User Performant File Access):
  Caching über die Einbettung der Hiperfiles in das DVS

Mit ADM-PFA-Caching wird die Verwendung der Konzepte, Methoden und Kommandos des Cache-Handler-Subsystems DAB (START-DAB-CACHING) bezeichnet, um sie von der Einbettung der Hiperfiles in das DVS, das sog. PFA-Konzept, begrifflich abzugrenzen. ADM-PFA-Caching wird im Handbuch zu „DAB“ [10] beschrieben.

BS2000/OSD unterstützt im Rahmen des PFA-Konzeptes die Cache-Medien Hauptspeicher und Globalspeicher (GS). Als Treibersoftware zur Bedienung dieser Medien (Cache-Handler) wird das Subsystem DAB benötigt:

- für das Medium Hauptspeicher auf allen BS2000/OSD-Servern
- für das Medium Globalspeicher auf S-Servern

Die Einbettung der Hiperfiles in das DVS wird zum einen dadurch erreicht, dass Pubsets Cache-Medien zugeordnet werden können, und zum anderen dadurch, dass der Benutzer seine Dateien durch Vergabe von performance-bezogenen Attributen zu Hiperfiles erklären kann.



Die Besonderheiten für den Einsatz der Cache-Medien Hauptspeicher und Globalspeicher sind ausführlich im Handbuch „DAB“ [10] beschrieben.
Die Beschreibung des Cache-Mediums Globalspeicher ist im Abschnitt „Globalspeicher (S-Server)“ auf Seite 136 zu finden.

Im vorliegenden Abschnitt sind diejenigen Teile des PFA-Konzeptes beschrieben, die aus Sicht der Systembetreuung relevant sind. Die Nutzung des PFA-Konzeptes durch den Benutzer ist im Handbuch „Einführung in das DVS“ [19] beschrieben. Dort finden sich u.a. die Abschnitte „Auswahl der Dateien“ und „Fehler beim Schließen eines Hiperfiles“.

Globalspeicher stehen nur auf S-Servern zur Verfügung.

Mit dem Kommando START-FILE-CACHING kann nachträglich für bereits geöffnete Dateien Caching gestartet werden. Über die Operanden PERFORMANCE und USAGE werden die Dateieigenschaften Performance-Wertigkeit und Cache-Modus bestimmt.

Voraussetzungen für das nachträgliche Zuschalten von Caching sind:

- dass der Aufrufer die Zugriffsberechtigung auf die Datei besitzt (Eigentümer der Datei oder die Systembetreuung)
- dass der Pubset, auf dem sich die Datei befindet,
  - eine gültige und aktivierte Cache-Zuordnung besitzt
  - lokal zugreifbar sein muss
- dass das Kommando am selben Server eingegeben werden muss, wenn der dem Pubset zugeordnete Cache lokal am Server betrieben wird
- dass dem Aufrufer die erforderliche Cache-Berechtigung von der Systembetreuung erteilt wurde

Ist die Datei nicht geöffnet oder wird sie bereits mit Cache bearbeitet, wird START-FILE-CACHING abgewiesen.

Das Kommando SHOW-DAB-CACHING informiert die Systembetreuung über die aktuell installierten Cache-Bereiche in den Cache-Medien Haupt- und Globalspeicher.

Das Kommando STOP-FILE-CACHING beendet das Caching für eine geöffnete Datei oder eine Datei, von der sich noch Daten im Cache befinden. Die noch im Cache befindlichen Daten werden zurückgeschrieben (Ausnahme: reiner Lese-Cache) und der Cache-Inhalt invalidiert. Zum Ausführen des Kommandos gelten die gleichen Voraussetzungen wie bei START-FILE-CACHING.

## 8.3.2 Dateiattribute von Hiperfiles

Zur Steuerung der Cache-Nutzung im Hiperfile-Konzept dienen die performance-bezogenen Dateiattribute PERFORMANCE, USAGE und DISK-WRITE:

- PERFORMANCE
  spezifiziert die gewünschte Performance-Güte bzgl. des Ein-/Ausgabe-Verhaltens bei Dateizugriffen
- USAGE
  beschreibt, ob die Performance nur für lesende, nur für schreibende oder beide Arten von Zugriffen gewünscht wird
- DISK-WRITE
  bestimmt, zu welchem Zeitpunkt der Dateibearbeitung mit Caching Datenkonsistenz für Schreiboperationen gegeben sein muss

**PFA bei SF-Pubsets**

Die performance-bezogenen Dateiattribute steuern bei SF-Pubsets die Nutzung der Caches: Eine Datei, für die eine erhöhte Performance-Anforderung besteht, wird automatisch mit Cache-Pufferung bearbeitet. Dabei legen die USAGE- und DISK-WRITE-Vorgaben und der dem Pubset zugeordnete Cache-Typ (VOLATILITY) fest, ob ein Lese- und/oder ein Schreib-Caching erfolgt.

Werden die Dateiattribute PERFORMANCE=*HIGH, USAGE=*READ-WRITE und DISK-WRITE=*IMMEDIATE für eine Datei gewählt, die auf einen SF-Pubset mit einem nicht-schreibsicheren Cache (z.B. Hauptspeicher-Cache) gelegt wird, so erfolgt die Cache-Zwischenpufferung nur lesend, da die Voraussetzungen für ein ausfallsicheres Caching der Schreib-I/Os nicht erfüllt sind. Es obliegt dem Benutzer selbst, sicherzustellen, dass die von ihm vorgegebenen Dateiattribute mit den Cache-Eigenschaften harmonieren.

Das Kommando SHOW-MASTER-CATALOG-ENTRY INF=*USER zeigt über die Ausgabezeile CACHE-MEDIUM an, ob ein nicht-flüchtiges Cache-Medium dem Pubset zugeordnet ist. Bei der Ausgabe NONVOLATILE ist dies der Fall; nur dann kommen die Dateiattribute USAGE=*WRITE und DISK-WRITE=*IMMEDIATE zur Wirkung.

Der Systembetreuer kann sich im Kommando SHOW-PUBSET-CACHE-ATTRIBUTES die komplette Cache-Konfiguration von SF-Pubsets anzeigen zu lassen.

Für SF-Pubsets, die keinen Cache zugeordnet haben, sind die Attribute PERFORMANCE, USAGE und DISK-WRITE ohne Relevanz. Legt ein Benutzer eine Datei z.B. auf einen im GS emulierten SF-Pubset ab, so ist das Performance-Verhalten bereits vollständig durch den Ablageort bestimmt.

Diese Beispiele zeigen, dass der Benutzer die Merkmale der verschiedenen Medien sehr genau kennen muss, wenn er die Dateiattribute selbst ändert, um ein gewünschtes Performance-Verhalten zu erreichen.



Der Systembetreuer sollte den Cache-Bereich eines Pubsets so konfigurieren, dass der Benutzer seine Dateien nicht selbst attributieren muss, um eine höhere Performance zu erreichen:
Für die Cache-Medien Hauptspeicher und GS sollte eine Einstellung getroffen werden, bei der die Cache-relevanten Dateien des Pubsets vom System selbst ermittelt werden (Operand CACHED-FILES=*BY-SYSTEM).

In diesen Fällen stellen die Standardwerte für die Dateiattribute sicher, dass ein Schreib-Caching nur in ausfallsicheren Medien (GS mit Batterie-Backup) erfolgt. Ein Lese-Caching wird auf jeden Fall durchgeführt.

### PFA bei SM-Pubsets

Die performance-bezogenen Dateiattribute können bei SM-Pubsets der Volume-Set-Selektion zur Bestimmung des zur Dateiablage am besten geeigneten Volume-Sets dienen:
Bei der Neuanlage einer Datei wird diese automatisch anhand ihrer Performance-Attribute auf ein Volume-Set gebracht, dessen Performance-Profil den Dateianforderungen am besten genügt.

Das Performance-Profil eines Volume-Sets ergibt sich aus dem Performance-Spektrum eines Volume-Sets und der Einschränkung bzgl. der Schreibsicherheit bei erhöhten Performance-Werten. Es kann durch den Operanden PERFORMANCE-ATTR im Kommando MODIFY-PUBSET-DEFINITION-FILE festgelegt und mit dem Kommando SHOW-PUBSET-DEFINITION-FILE abgefragt werden. Die Komplexität ist notwendig, da das Performance-Profil die unterschiedlichen Cache- und Plattentypen zu charakterisieren hat. Der Ausprägung der performance-relevanten Attribute eines Volume-Sets liegen folgende Überlegungen zu Grunde:

Ein mit einem GS-Cache ausgestatteter Volume-Set bietet bei Dateibearbeitung ohne Cache normale Performance (STD), bei Dateibearbeitung mit Cache erhöhte Performance (HIGH bzw. VERY-HIGH). Ein Volume-Set mit Cache verfügt also über ein Performance-Spektrum (von STD über HIGH bis zu VERY-HIGH).

Dem wird dadurch Rechnung getragen, dass dem Attribut PERFORMANCE (Unteroperand von PERFORMANCE-ATTR) eine Liste von Werten zugewiesen werden kann. Allerdings kann bei nicht-ausfallsicheren Caches die erhöhte Performance bzgl. Schreiboperationen nur bei Dateibearbeitungen geboten werden, bei denen keine Datenkonsistenz nach jeder Schreiboperation gefordert wird. Um diese Einschränkung bzgl. erhöhter Performance auszudrücken, wird als weiteres performance-relevantes Attribut WRITE-CONSISTENCY (Unteroperand von PERFORMANCE-ATTR) eingeführt:

Der Wert WRITE-CONSISTENCY=*IMMEDIATE bedeutet, dass auf dem Volume-Set erhöhte Performance uneingeschränkt geboten werden kann, WRITE-CONSISTENCY=*BY-CLOSE bringt zum Ausdruck, dass erhöhte Performance bei Schreiboperationen nur dann genutzt werden kann, wenn eine Datenkonsistenz zum CLOSE-Zeitpunkt ausreicht und dies auch so angefordert wird.

Es obliegt dem Systembetreuer, die dem Performance-Profil eines Volume-Sets adäquaten Hardware- bzw. Cache-Konfigurationen mit Hilfe des Kommandos MODIFY-PUBSET-CACHE-ATTRIBUTES (EDIT-PUBSET-CACHE-ATTRIBUTES) bereitzustellen, bzw. den Volume-Set aus GS-Volumes zusammenzustellen.

Die folgende Tabelle gibt einen Überblick über den Zusammenhang zwischen Performance-Profil eines Volume-Sets und der jeweiligen Volume-Set-Konfiguration:

| **Volume-Set-Konfiguration** | **empfohlenes Performance-Profil** | |
|---|---|---|
| | **Performance-Spektrum (PERFORMANCE)** | **Schreibsicherheit (WRITE-CONSISTENCY) bei erhöhter Performance (PERF=HIGH/VERY-HIGH)** |
| Volume-Set, der aus Volumes ohne besondere Performance-Eigenschaften besteht und dem kein Cache zugeordnet ist | STD | ohne Bedeutung |
| Volume-Set, dem ein Hauptspeicher als Cache-Medium zugeordnet ist | STD, HIGH | nicht gegeben (BY-CLOSE) |
| Volume-Set, dem ein flüchtiger Globalspeicher zugeordnet ist (VOLATILITY=*YES) | STD, HIGH, VERY-HIGH | nicht gegeben (BY-CLOSE) |
| Volume-Set, dem ein nicht-flüchtiger Globalspeicher zugeordnet ist (VOLATILITY=*NO) | STD, HIGH, VERY-HIGH | gegeben (IMMEDIATE) |
| Volume-Set, der aus im Globalspeicher emulierten Volumes besteht | VERY-HIGH | gegeben (IMMEDIATE) |
| Volume-Set, dessen Volumes an eine Cache-Steuerung angeschlossen sind | STD, HIGH | gegeben (IMMEDIATE) |

Tabelle 20: Empfehlungen für Performance-Profile

*Beispiel*

Gegeben ist ein Volume-Set mit einem schreibsicheren GS-Cache.

Es ist folgende Zuweisung zu empfehlen:

```
PERFORMANCE=(*STD,*HIGH,*VERY-HIGH)
WRITE-CONSISTENCY=*IMMEDIATE
```

Mit SHOW-PUBSET-CACHE-ATTRIBUTES ...,VOLUME-SET=*ALL/<cat-id 1..4> kann sich der Systembetreuer die eingestellten Cache-Konfigurationen der Volume-Sets eines SM-Pubset anzeigen lassen.

### 8.3.3 Die Cache-Zuordnung von Pubsets

Einem Cache-Medium können ein oder mehrere SF-Pubsets bzw. Volume-Sets von SM-Pubsets zugeordnet werden. Mit dem Kommando MODIFY-PUBSET-CACHE-ATTRIBUTES (EDIT-PUBSET-CACHE-ATTRIBUTES) kann die Systembetreuung u.a. festlegen, welcher Cache-Anteil auf dem Cache-Medium für diesen Pubset reserviert werden soll. Die Zuordnung wird automatisch beim Importieren des Pubsets (IMPORT-PUBSET) oder bei der Inbetriebnahme eines Volume-Sets für ein SM-Pubset (MODIFY-PUBSET-PROCESSING) wirksam, kann aber auch dynamisch im laufenden Pubset-Betrieb (START-PUBSET-CACHING) aktiviert werden.

Die im Zusammenhang mit einer Cache-Konfiguration relevanten Attribute FORCE-IMPORT und SIZE-TOLERANCE beziehen sich stets auf den ganzen SM-Pubset und sind durch das Kommando MODIFY-PUBSET-CACHE-ATTRIBUTES für den SM-Pubset einzustellen.

Sowohl durch Exportieren des Pubsets (Kommando EXPORT-PUBSET) oder durch die Außerbetriebnahme eines Volume-Sets eines SM-Pubsets (MODIFY-PUBSET-PROCESSING) als auch explizit durch das Kommando STOP-PUBSET-CACHING wird der Speicherbereich im Cache-Medium aufgelöst. Auskunft über Cache-Hitraten kann die Systembetreuung aus der Analyse des openSM2-Messreports PFA erhalten, der Informationen über die Nutzung der Cache-Medien liefert.

Mit dem Kommando SHOW-PUBSET-CACHE-ATTRIBUTES kann sich die Systembetreuung über die aktuell gültige und aktivierte Cache-Zuordnung, die Cache-Medien, die Betriebsarten und Größenvereinbarungen informieren. Darüberhinaus kann detaillierte Auskunft über die Konfiguration der einzelnen PFA-Cache-Bereiche mit dem Kommando SHOW-CACHE-CONFIGURATION eingeholt werden.

Die Systembetreuung kann die Cache-Nutzung für alle Dateien eines Pubsets und alle Benutzer pauschal ermöglichen, indem sie im Kommando MODIFY-PUBSET-CACHE-ATTRIBUTES die Angabe CACHED-FILES=*ALL macht. Für die Cache-Medien GS und Hauptspeicher ist darüber hinaus die Einstellung CACHED-FILES=*BY-SYSTEM möglich, bei der die Cache-relevanten Dateien vom System selbst bestimmt und bzgl. ihrer Cache-Nutzung überwacht werden. Damit wird eine optimale Performance der Dateien eines Pubsets gewährleistet (sog. AutoDAB).
Dabei werden für jede Datei des Pubsets (mit Ausnahme einiger System-Dateien, z.B. Datei- und Benutzerkatalog) zunächst Standardwerte gesetzt:

- „Schreib-Lese-Cache“ für das Cache-Medium GS
- „Lese-Cache“ für das flüchtige Cache-Medium Hauptspeicher
- mit der Datenbereichsangabe AREA=*BY-SYSTEM (Kommando START-DAB-CACHING): für temporäre Dateien im Hauptspeicher ebenfalls „Schreib-Lese-Cache“

Diese Voreinstellung kann vom Benutzer ohne zusätzliche Berechtigung geändert werden. Alternativ dazu besteht die Möglichkeit, bestimmten Benutzern eine Cache-Nutzung zu ermöglichen, indem ihnen das Attributieren ihrer Dateien erlaubt wird.

Die Aufgabe der Systembetreuung ist, Benutzerkennungen zu berechtigen, Dateien über Caches zu bearbeiten (Hiperfiles) und zusätzlich bei SM-Pubsets den Benutzern entsprechend große Kontingente für Hiperfiles (HIGH-PERF-SPACE und VERY-HIGH-PERF-SPACE) zuzuweisen.

Die Berechtigung wird durch einen entsprechenden Eintrag im Benutzerkatalog des Pubsets mit dem Kommando ADD-USER bzw. MODIFY-USER (Operand DMS-TUNING-RESOURCES) vergeben und kennzeichnet das für die Dateien des Benutzers jeweils höchstzulässige Performance-Attribut. Zusätzlich steht dafür das Kommando MODIFY-PUBSET-USER-ATTRIBUTES zu Verfügung, mit dem auch die jeweiligen Kontingente für SM-Pubsets zugewiesen werden können.

Die Systembetreuung kann dem einzelnen Benutzer folgende Caching-Berechtigungen verleihen:

- NONE
  Der Benutzer erhält keine Berechtigung, Dateien über Caches zu bearbeiten.
- CONCURRENT-USE
  Der Benutzer ist berechtigt, Dateien über einen Cache zu bearbeiten, steht dabei aber in Konkurrenz zu allen anderen Benutzern mit der gleichen Berechtigung. Bei knappem Speicherplatz im Cache-Medium können also Teile der Benutzerdatei wieder ausgelagert werden.
- EXCLUSIVE-USE
  Ber Benutzer ist berechtigt, Dateien über einen Cache exklusiv zu bearbeiten. Auch bei knappem Speicherplatz versucht das System genau dann die Datei des Benutzers immer vollständig im Cache-Medium zu halten, wenn der Benutzer dies durch ein entsprechendes Datei-Attribut anfordert.

Systembetreuung und Benutzer können sich mit dem Kommando SHOW-USER-ATTRIBUTES über die pubset-spezifischen Berechtigungen und Kontingente der ausgewählten Benutzerkennung informieren.



Der Home-Pubset kann nicht in den von DAB verwalteten Cache-Medien mit PFA-Caching gepuffert werden. Im Falle eines Systemausfalls ist die Rekonstruktion von Schreibdaten nicht gewährleistet, weil zum Zeitpunkt des Imports des Home-Pubsets das Subsystem DAB noch nicht zur Verfügung steht. Das Einrichten eines PFA-Cache-Bereichs für den Home-Pubset wird deshalb abgelehnt.

## 8.3.4 Beispiele

Die folgenden beiden Beispiele sollen anhand des Cache-Mediums Globalspeicher (GS) den Ablauf aus Sicht der Systembetreuung für das Einrichten von PFA-Caches zusammenfassen.

**Beispiel 1: SF-Pubset**

Dateien des importierten Daten-Pubsets DAT1 (SF-Pubset) sollen im Globalspeicher gepuffert werden.

```
/SHOW-GS-STATUS *ALL ————————————————————————————————————————————————————  (1)

SUBSYSTEM-PARAMETER:
GS-USAGE=LOCAL

GS-UNITS:
GS-UNIT  BEGIN    END   SIZE   FREE CONTIGUOUS  STATE
   1         0    255    256    190        190  ATTACHED

PARTITIONS:
PART-ID  ACCESS DATA  MODE GSU  SIZE BEGIN   END ATTACH-DUAL HOST/SYSID
GSV01    SHARED VALID MONO 1      64     2    65                    /99

FREE MONO GS SPACE:
GS-UNIT    SIZE     BEGIN       END
   1        190        66       255
NO (MORE) FREE DUAL GS-SPACE AVAILABLE

/CREATE-GS-PARTITION PARTITION-ID=DAB1HPVS,SIZE=90,MODE=*MONO(GS-UNIT=1)  (2)
/MODIFY-PUBSET-CACHE-ATTRIBUTES PUBSET=DAT1,CACHE-MEDIUM=*GLOBAL-STORAGE(
        VOLATILITY=*NO,GLOBAL-STORAGE-UNIT=*MONO(UNIT=1),
        CACHE-FILES=*BY-SYSTEM),
        CACHE-SIZE=90(DIMENSION=MEGABYTE) ————————————————————————————————  (3)
/SHOW-PUBSET-CACHE-ATTRIBUTES PUBSET=DAT1 ————————————————————————————————  (4)
-----------------------------------------------------------------------------
PUBSET DAT1: SINGLE-FEATURE, ACC

---- CACHE CONFIGURATION    --------- + ---- DEFINED ---- + ---- CURRENT ----
 CACHE MEDIUM                         | GLOBAL STORAGE    | NONE
 CACHE SIZE                           | 90           MB   |
 CACHED FILES                         | BY SYSTEM         |
 VOLATILITY       (GLOBAL STORAGE)    | NO                |
 GS-UNIT          (GLOBAL STORAGE)    | MONO (1)          |
 PREFETCH         (CONTROLLER    )    | NO                |
 SEGMENT SIZE     (NOT CONTROLLER)    | 4            KB   |
 FORCE OUT        (NOT CONTROLLER)    | AT LOW FILLING    |
-----------------------------------------------------------------------------
```

```
/START-PUBSET-CACHING PUBSET=DAT1 ------------------------------------- (5)
/SHOW-CACHE-CONFIGURATION CACHE-ID=DAT1 --------------------------------- (6)

   CACHE-CONFIGURATION IN CACHE-MEDIUM GS OF CACHE-METHOD DAB:

   PARTITION-CONFIGURATION-RECORD:
   PARTITION-ID    SIZE   FREE-SIZE   GS-UNIT     STATUS         #BUFFER
   DAB1HPVS        90MB         0MB      1          PFA               1

   BUFFER-CONFIGURATION-RECORD:
   CACHE-ID        SIZE  SEGMENT-SIZE  IN-USE     STATUS         FORCE-OUT
   DAT1            90MB       4KB        0%     CONNECTED    AT-LOW-FILLING
```

(1) Mit dem Kommando SHOW-GS-STATUS wird der Belegungszustand des Globalspeichers ermittelt.

(2) Mit dem Kommando CREATE-GS-PARTITION wird im Globalspeicher die DAB-Home-Partition DAB1HPVS der Größe 90 MB in der GS-Unit 1 angelegt.

(3) Mit dem Kommando MODIFY-PUBSET-CACHE-ATTRIBUTES wird für den SF-Pubset DAT1 eine Cache-Konfiguration definiert. Es soll ein Cache-Bereich mit der Größe 90 MB in der GS-Unit 1 angelegt werden.

(4) Mit dem Kommando SHOW-PUBSET-CACHE-ATTRIBUTES kann die unter Punkt (3) erfolgte Festlegung der Cache-Konfiguration überprüft werden.

(5) Mit dem Kommando START-PUBSET-CACHING wird der zuvor definierte Cache-Bereich in Betrieb genommen (aktiviert).

(6) Mit dem Kommando SHOW-CACHE-CONFIGURATION kann der aktivierte Cache-Bereich überprüft werden. Da keine GS-Partition mit dem Namen DAB1DAT1 existierte, wurde der Cache-Bereich in die GS-Partition DAB1HPVS gelegt.

### Beispiel 2: Volume-Set eines SM-Pubsets

Der Volume-Set VS01 des importierten Daten-Pubset SMS1 (SM-Pubset) soll zusätzlich im GS gepuffert werden. Dem Benutzer USER2 soll für diesen Pubset die Berechtigung zum Caching gegeben werden und ihm sollen entsprechende Kontingente zugewiesen werden.

```
/SHOW-GS-STATUS *ALL ────────────────────────────────────────────────── (1)

SUBSYSTEM-PARAMETER:
GS-USAGE=LOCAL

GS-UNITS:
GS-UNIT  BEGIN    END   SIZE   FREE CONTIGUOUS  STATE
   1         0    255    256    100        100  ATTACHED

PARTITIONS:
PART-ID  ACCESS DATA  MODE GSU  SIZE BEGIN   END ATTACH-DUAL HOST/SYSID
GSV01    SHARED VALID MONO 1      64     2    65                    /99
DAB1HPVS EXCL   VALID MONO 1      90    66   155                     /99

FREE MONO GS SPACE:
GS-UNIT    SIZE     BEGIN       END
   1        100       156       255
NO (MORE) FREE DUAL GS-SPACE AVAILABLE

/CREATE-GS-PARTITION PARTITION-ID=DAB1SMS1,SIZE=50,MODE=*MONO(GS-UNIT=1)  (2)
/MODIFY-PUBSET-CACHE-ATTRIBUTES PUBSET=SMS1,PUBSET-TYPE*SYSTEM-MANAGED(
        VOLUME-SET=VS01),CACHE-MEDIUM=*GLOBAL-STORAGE(
        CACHED-FILES=*BY-USER,
        VOLATILITY=*NO,GLOBAL-STORAGE-UNIT=*MONO(UNIT=1),
        CACHE-SEGMENT-SIZE=*4KB,FORCE-OUT=*AT-LOW-FILLING),
        CACHE-SIZE=50(DIMENSION=MEGABYTE) ───────────────────────────── (3)
/SHOW-PUBSET-CACHE-ATTRIBUTES PUBSET=SMS1,VOLUME-SET=VS01 ───────────── (4)
------------------------------------------------------------------------------
PUBSET SMS1: SYSTEM-MANAGED, CTL-SET=(VS00,STDDISK), ACC

---- VOLUME-SET INFORMATION --------------------------------------------------
VOLUME-SET VS01: STDDISK, NORMAL-USE

---- CACHE CONFIGURATION    --------- + ---- DEFINED ---- + ---- CURRENT ----
 CACHE MEDIUM                        | GLOBAL STORAGE    | NONE
 CACHE SIZE                          | 50            MB  |
 CACHED FILES                        | BY USER           |
 VOLATILITY       (GLOBAL STORAGE)   | NO                |
 GS-UNIT          (GLOBAL STORAGE)   | MONO (1)          |
 PREFETCH         (CONTROLLER    )   | NO                |
 SEGMENT SIZE     (NOT CONTROLLER)   | 4             KB  |
 FORCE OUT        (NOT CONTROLLER)   | AT LOW FILLING    |
------------------------------------------------------------------------------
```

```
/MODIFY-USER-PUBSET-ATTRIBUTES USER-IDENTIFICATION=USER2,
        PUBSET=SMS1,RIGHTS=*PARAMETERS(
        DMS-TUNING-RESOURCES=*CONCURRENT-USE),PERM-SPACE-LIMITS=*PARAMETERS(
        HIGH-PERF-SPACE=*AUTOMATIC-ADAPT) ——————————————————————————————   (5)
/START-PUBSET-CACHING PUBSET=SMS1.
        PUBSET-TYPE=*SYSTEM-MANAGED(VOLUME-SET=VS01) ———————————————————   (6)
/SHOW-CACHE-CONFIGURATION CACHE-ID=VS01 ————————————————————————————————   (7)

CACHE-CONFIGURATION IN CACHE-MEDIUM GS OF CACHE-METHOD DAB:

PARTITION-CONFIGURATION-RECORD:
PARTITION-ID    SIZE   FREE-SIZE   GS-UNIT     STATUS         #BUFFER
DAB1SMS1        50MB         0MB      1         PFA                1

BUFFER-CONFIGURATION-RECORD:
CACHE-ID        SIZE  SEGMENT-SIZE  IN-USE     STATUS         FORCE-OUT
VS01            50MB       4KB        0%     CONNECTED    AT-LOW-FILLING
```

(1) Mit dem Kommando SHOW-GS-STATUS wird der Belegungszustand des Globalspeichers ermittelt.

(2) Mit dem Kommando CREATE-GS-PARTITION wird im Globalspeicher die DAB-Partition DAB1SMS1 der Größe 50 MB, reserviert für den SM-Pubset SMS1, in der GS-Unit 1 angelegt.

(3) Mit dem Kommando MODIFY-PUBSET-CACHE-ATTRIBUTES wird für den Volume-Set VS01 des SM-Pubsets SMS1 eine Cache-Konfiguration definiert. Es soll ein Cache-Bereich mit der Größe 50 MB in der GS-Unit 1 angelegt werden.

(4) Mit dem Kommando SHOW-PUBSET-CACHE-ATTRIBUTES kann die unter Punkt (3) erfolgte Festlegung der Cache-Konfiguration überprüft werden.

(5) Mit dem Kommando MODIFY-USER-PUBSET-ATTRIBUTES wird der Benutzerkennung USER2 für den Pubset SMS1 die Berechtigung erteilt, ihren Dateien das Performance-Attribut PERFORMANCE=HIGH zu geben. Ihr wird ein Kontingent (HIGH-PERF-SPACE) zum Einrichten von hoch performanten, permanenten Dateien zugewiesen.

(6) Mit dem Kommando START-PUBSET-CACHING wird der zuvor definierte Cache-Bereich in Betrieb genommen (aktiviert).

(7) Mit dem Kommando SHOW-CACHE-CONFIGURATION kann der aktivierte Cache-Bereich überprüft werden.

### 8.3.5 Die Cache-Medien Haupt- und Globalspeicher

Die Cache-Medien werden vom Cache-Handler **DAB** (Softwareprodukt) verwaltet.

Der Globalspeicher bietet über die Option zweier ausfallunabhängiger GS-Units und über die Möglichkeit der Wiederherstellung von DAB-Cache-Bereichen nach einem Systemausfall komplexere Funktionen als der Hauptspeicher.

Beim Einsatz der Cache-Medien sollte generell das Handbuch „DAB“ [10] zum besseren Verständnis zu Rate gezogen werden. Es enthält auch eine ausführliche Beschreibung der Besonderheiten des Globalspeichers als Cache-Medium.

| Kommando | Bedeutung | Bereich |
|---|---|---|
| FORCE-DESTROY-CACHE | Die Auflösung eines vorhandenen PFA-Cache-Bereichs erzwingen | PFA-Caching |
| FORCE-STOP-DAB-CACHING | Auflösung eines vorhandenen ADM-PFA-DAB-Cache-Bereiches erzwingen | ADM-PFA-Caching |
| MODIFY-DAB-CACHING | Parameter eines DAB-Cache-Bereichs dynamisch ändern | ADM-PFA-Caching und PFA-Caching |
| MODIFY-DAB-PARAMETERS | DAB-Subsystem-Parameter dynamisch ändern | ADM-PFA-Caching und PFA-Caching |
| SHOW-CACHE-CONFIGURATION | Konfiguration der PFA-Cache-Bereiche anzeigen | PFA-Caching |
| SHOW-DAB-CACHING | Informationen über die aktuelle DAB-Konfiguration einholen | ADM-PFA-Caching und PFA-Caching |
| START-DAB-CACHING | ADM-PFA-DAB-Cache-Bereiche anlegen | ADM-PFA-Caching |
| STOP-DAB-CACHING | ADM-PFA-DAB-Cache-Bereiche auflösen | ADM-PFA-Caching |

Tabelle 21: Übersicht über die DAB-Kommandos

In allen von DAB verwalteten Cache-Medien ist für jeden Cache-Bereich die Segmentgröße variabel einstellbar. DAB führt grundsätzlich für jedes Cache-Medium eine Segmentverwaltung durch: bei einer Einlagerung von Daten im Cache werden immer ganze Segmente (Plattenbereiche von 64 KB je nach Segmentgröße bzw. bei automatischem Caching) bewegt. Die Wahl großer Segmente ist sinnvoll bei Anwendungen mit vorwiegend sequenzieller Verarbeitung (oder allgemeiner mit hoher räumlicher Lokalität). Durch die Wahl kleiner Segmente kann ein „Record Level Cache“ für Anwendungen mit Direkt-Zugriff eingesetzt werden. Dies wirkt sich dann äußerst positiv aus, wenn die Anwendungen in der Regel eine nur geringe räumliche, dafür aber eine hohe zeitliche Lokalität (Mehrfach-Zugriff auf den gleichen Block innerhalb eines Zeitintervalls) aufweisen. Die in diesem Fall negativen Effekte der Einlagerungen großer Segmente in den Cache werden vermieden und die Cache-Bereichs-Größe kann entscheidend verringert werden.

Die Auswahl der Segmentgröße muss nicht von der Systembetreuung getroffen werden. Für die Funktionen des automatisierten Cachings (AutoDAB) wird von DAB immer der für die Anwendung passende Prefetch durchgeführt.

AutoDAB bietet folgende Funktionen:

- Durch intelligente Caching-Algorithmen werden für das Caching geeignete Dateien automatisch ausgewählt.
- Für die ausgewählten Dateien wird der zu ihrem Zugriffsprofil passende Prefetch-Faktor eingestellt.
- Die gepufferten Dateien werden bzgl. ihrer Cache-Nutzung zyklisch überwacht.

### 8.3.6 Caching verschlüsselter Dateien

DAB unterstützt das Lese-Caching von verschlüsselten und unverschlüsselten Dateien, nicht jedoch das Schreib-Caching von verschlüsselten Dateien.
Lese-Caching ist für verschlüsselte Dateien günstig, um die durch die Entschlüsselung längerten Zugriffszeiten zu verkürzen.

Die Daten werden im Cache-Speicher in entschlüsselter Form gespeichert. Damit fällt bei allen Datenzugriffen, die nur einen Datentransfer in oder aus dem Cache-Speicher erfordern, die durch die Verschlüsselungsvorgänge entstehende Pfadverlängerung nicht mehr an. Die Ver- bzw. Entschlüsselung wird nur beim Datentransfer zwischen Cache-Speicher und Plattenspeicher erforderlich.

Vorteile des Lese-Cachings:

1. Das Datenvolumen, das entschlüsselt werden muss, wird reduziert.
2. Die Anzahl der kritischen Datentransfers wird reduziert.
3. Die durch die Crypto-Verfahren entstehende Verlängerung der Lesezugriffe durch den größeren I/O-Blockungsfaktor bei der Einlagerung von Daten im Cache wird gemindert.

Schreib-Caching von verschlüsselten Dateien wird abgewiesen.
Für das Lese-Caching von verschlüsselten Dateien hat der Benutzer die gleichen Schnittstellen und Funktionen zur Verfügung wie für das Caching unverschlüsselter Dateien.

## 8.4 Versenden von BS2000-Dateien per E-Mail

Mit BS2000/OSD können Sie im BS2000-Betrieb auf einfache Weise BS2000-Dateien oder Bibliothekselemente per E-Mail versenden.

Als Voraussetzung werden in den Benutzerkatalog des Home-Pubsets für die Benutzerkennungen entsprechende E-Mail-Adressen eingetragen, siehe nächste Seite. Zudem muss der Mail-Service des Software-Produkts interNet Services installiert sein.

Mit dem Kommando MAIL-FILE (siehe Handbuch „Kommandos“ [27] oder dem Makro MAILFIL (siehe Handbuch „DVS-Makros“ [20]) können Sie BS2000-Dateien (auch SYSLST und SYSOUT) an die angegebene Benutzerkennung versenden. Diese E-Mail wird an eine ausgewählte oder an alle Empfängeradressen versandt, die im Benutzereintrag der Benutzerkennung eingetragen sind.

Bei Versand an die eigene Benutzerkennung wird nach Möglichkeit nur diejenige E-Mail-Adresse als Empfänger ausgewählt, die den Jobnamen am Anfang oder nach einem Punkt im vorderen Teil der E-Mail-Adresse enthält (z.B. Jobnamen `MAX` oder `MUSTER`, E-Mail-Adresse im Benutzereintrag: `Max.Mustermann@xyz.net`). Siehe die detaillierte Beschreibung beim Kommando MAIL-FILE im Handbuch „Kommandos“ [27].

Für die zu versendende Text-Datei wird der Zeichensatz (Dateiattribut CCS-NAME) berücksichtigt und beim Versand eine Konvertierung in einen Zeichensatz der offenen Systemwelt durchgeführt. Sie können der E-Mail einen „Betreff“ (SUBJECT) hinzufügen und festlegen, ob die Datei nach erfolgreichem Versand gelöscht werden soll.

Als Absender der E-Mail wird bei erfolgreicher Auswahl einer E-Mail-Adresse über den Jobnamen diese E-Mail-Adresse verwendet, sonst die erste E-Mail-Adresse der Ablaufkennung. Zusätzlich wird für mögliche Zustellfehler die erste E-Mail-Adresse der Kennung TSOS mitgegeben. Auf diese Weise erhält der Administrator bei falschen Empfangsadressen eine sogenannte „Bounce-Mail“ und kann daraufhin die entsprechenden E-Mail-Adressen in der Benutzerverwaltung korrigieren.

Der eigentliche Versand der E-Mail erfolgt mit dem Mail-Sender (Schnittstelle SEND-MAIL) der Software-Produkts interNet Services.

Folgende Dateien können Sie mit diesem E-Mail-Konzept ebenfalls versenden:

- MAIL-FILE für Systemdateien SYSLST und SYSOUT
  In den Kommandos EXIT-JOB, CANCEL-JOB und ENTER-PROCEDURE können Sie die Systemausgaben explizit mit MAIL-FILE versenden (anstelle der SPOOL-Ausgabe). Weitere Informationen finden Sie im Handbuch „Kommandos“ [27] bei den entsprechenden Kommandos.
- HSMS-Reports mit HSMS, siehe Handbuch „HSMS“ [24].
- Ausgabedateien von MAREN, siehe Handbuch „MAREN“ [31].

### E-Mail-Adressen in den Benutzereinträgen

E-Mail-Adressen werden mit dem Operanden EMAIL-ADDRESS der Kommandos ADD-USER und MODIFY-USER-ATTRIBUTES im Benutzereintrag der BS2000-Benutzerkennungen eingetragen. (Das Attribut MAIL-ADDRESS bleibt unverändert.)

Für die Benutzerkennung TSOS muss eine E-Mail-Adresse eingetragen sein.

Mehrere E-Mail-Adressen einer Benutzerkennung werden als eine Zeichenfolge, getrennt durch Kommata, angegeben. Mit dem Kommando SHOW-USER-ATTRIBUTES bzw. dem Makro SRMUINF werden die eingetragenen E-Mail-Adressen ausgegeben.

Nach Eintrag neuer E-Mail-Adressen sollte die Systembetreuung zur Kontrolle eine Test-Mail an die zugehörige Benutzerkennung versenden.

### Ausgabe von SYSLST und SYSOUT mit MAIL-FILE

MAIL-FILE kann die aktuellen Systemdateien SYSLST und SYSOUT ausgeben. In den Kommandos EXIT-JOB, CANCEL-JOB und ENTER-PROCEDURE (siehe Handbuch „Kommandos“ [27]) kann für die Systemausgabe im entsprechenden Ausgabeoperanden der Wert `MAIL` anstelle von `PRINT` angegeben werden. Außerdem kann für diese Kommandos und für das Kommando LOGOFF der Standardwert der Systemausgabe im Systemparamter SSMOUT eingestellt werden auf `PRINT` oder `MAIL`.
Siehe Systemparameter „SSMOUT“ auf Seite 761.

Wenn bei Job-Ende kein E-Mail-Versand möglich ist, dann werden Systemdateien, die gemäß den Einstellungen als E-Mail verschickt werden müssten, ersatzweise wie bisher über SPOOL ausgegeben.

### Fehleranalyse im Mail-Service des Software-Produkts interNet Services

Der Systemverwalter kann über die Konfigurationsdatei des Software-Produkts interNet Services und über das Kommando MODIFY-MAIL-SERVICE-PARAMETER Einstellungen für Logging und Trace von E-Mails treffen. Logging- und Trace-Datei enthalten nähere Informationen zum Versand von E-Mails, insbesondere in Fehlerfällen.

## 8.5 Unicode in BS2000/OSD

Unicode fasst alle weltweit bekannten Textzeichen in einem einzigen Zeichensatz zusammen. Zudem ist Unicode unabhängig von unterschiedlichen Herstellern, Systemen und Ländern.

Mit der Unicode-Unterstützung im BS2000/OSD werden die in BS2000/OSD-Systemen verfügbaren EBCDIC-Zeichensätze um zusätzliche Zeichen erweitert, die im europäischen Sprachraum benötigt werden. Dem Anwender wird die Programmier- und Ablaufumgebung zur Verfügung gestellt, die er braucht, um seine bestehenden Anwendungen um Unicode-Datenfelder zu erweitern. Hierzu wird eine entsprechende Software-Konfiguration unter BS2000/OSD-BC bereitgestellt. Das Subsystem XHCS zur Unicode-Unterstützung in BS2000/OSD wird standardmäßig geladen.

Allgemeine Informationen zu Unicode entnehmen Sie bitte der Internetseite des Unicode-Konsortiums: *http://www.unicode.org/standard/translations/german.html*.

Nähere Hinweise zum Einsatz von Unicode in BS2000/OSD finden Sie im Handbuch „Unicode im BS2000/OSD“ [60].

# 9 Pubset-Verwaltung

Pubsets (Public-Volume-Sets) sind Sätze gemeinschaftlicher Platten und in BS2000/OSD neben den privaten Datenträgern der Ablageort für Dateien. Ein herausragendes Merkmal des Pubsets ist, dass neben den Dateien selbst auch alle für die Dateiverwaltung erforderlichen Metadaten (Dateikatalog, Benutzerkatalog, usw.) in ihm enthalten sind.

Es gibt Single-Feature-Pubsets (SF-Pubsets) und System-Managed-Pubsets (SM-Pubsets).



BS2000/OSD ab V9.0 bedient den so genannten Net-Storage im Rahmen der Pubsets, siehe Kapitel „Net-Storage-Verwaltung“ auf Seite 429.

Die folgenden Abschnitte befassen sich mit der Beschreibung des Objekts Pubset und dessen Verwaltung. Zu Beginn des Kapitels werden das Pubset-Konzept sowie die Pubset-Typen SF- und SM-Pubset erläutert.
Die folgenden Abschnitte beschäftigen sich mit den verschiedenen Pubset-Eigenschaften und deren Adminstration (z.B. Inbetriebnahme, Rekonfiguration, Überwachung).
Anschließend wird der Shared-Pubset beschrieben und wie Hardware-Fehler auf Pubsets erfasst und beseitigt werden können.
Ein weiterer Abschnitt beschreibt, wie bei Einsatz von Plattenspeichersystemen mit lokalen oder entfernten Replikationsfunktionen und des Software-Produkts SHC-OSD durch Erstellung von Standby-Pubsets die Datensicherheit und die Verfügbarkeit des Systems erhöht werden können.

Erläuterungen zur Verwaltung des SYSEAM-Speicherplatzes und zu SCA (Speed Catalog Access) schließen das Kapitel ab.

| **Kommando** | **Bedeutung** |
|---|---|
| ADD-MASTER-CATALOG-ENTRY | Eintrag im Katalogverzeichnis MRSCAT erzeugen |
| ADD-USER | Eintrag im Benutzerkatalog erstellen und Pubset-Zugriff festlegen |
| CANCEL-PUBSET-EXPORT | Exportieren eines Pubsets abbrechen |
| CANCEL-PUBSET-IMPORT | Importieren eines Pubsets abbrechen |
| CHECK-PUBSET-MIRRORS | Homogenität der Pubset-Spiegelung prüfen |
| EXPORT-PUBSET | Zuvor importierten Pubset exportieren |
| FORCE-PUBSET-EXPORT | Exportieren eines Pubsets erzwingen |

Tabelle 22: Kommandoübersicht zur Pubset-Verwaltung (Teil 1 von 3)

| Kommando | Bedeutung |
|---|---|
| IMPORT-PUBSET | Importieren eines Pubsets und Behandlung des Benutzerkataloges festlegen |
| MODIFY-MASTER-CATALOG-ENTRY (EDIT-MASTER-CATALOG-ENTRY) | Eintrag im Katalogverzeichnis MRSCAT ändern |
| MODIFY-PUBSET-CACHE-ATTRIBUTES (EDIT-PUBSET-CACHE-ATTRIBUTES) | PFA-Cache-Konfiguration für einen Pubset ändern |
| MODIFY-PUBSET-DEFINITION-FILE (EDIT-PUBSET-DEFINITION-FILE) | Definition eines SM-Pubsets ändern |
| MODIFY-PUBSET-PROCESSING | Zusammensetzung eines Pubsets ändern |
| MODIFY-PUBSET-RESTRICTION | Nutzungseinschränkungen für einen Pubset ändern |
| MODIFY-PUBSET-SPACE-DEFAULTS (EDIT-PUBSET-SPACE-DEFAULTS) | Standardwerte der Speicherplatzverwaltung ändern |
| MODIFY-PUBSET-SPACEPRO-OPTIONS | SPACEPRO-Optionen ändern |
| MODIFY-SPACEPRO-PARAMETERS | SPACEPRO-Parameter ändern |
| MODIFY-SPACE-SATURATION-LEVELS (EDIT-SPACE-SATURATION-LEVELS) | Schwellwerte für Speicherplatzsättigung ändern |
| MODIFY-USER-ATTRIBUTES | Eintrag im Benutzerkatalog ändern |
| MODIFY-USER-PUBSET-ATTRIBUTES | SM-Pubset-Attribute einer Benutzerkennung ändern |
| MOVE-SPACEPRO-DISK | SPACEPRO- oder Pool-Pubset erweitern oder reduzieren |
| REMOVE-MASTER-CATALOG-ENTRY | Eintrag im Katalogverzeichnis MRSCAT löschen |
| REMOVE-PUBSET-LOCK | Pubset-Lock entfernen |
| RESUME-PUBSET-RECONFIGURATION | Pubset-Rekonfigurationsauftrag ordnungsgemäß beenden |
| SET-PUBSET-ATTRIBUTES | Charakteristiken eines Pubsets festlegen |
| SET-SPACE-SATURATION-LEVEL | Sättigungsstufen für die Ausnutzung des Speicherplatzes auf einem SF-Pubset festlegen<br>(wird nur noch aus Kompatibilitätsgründen unterstützt) |
| SHOW-MASTER-CATALOG-ENTRY | Informationen über Zustand, Belegung und Zugreifbarkeit eines Pubsets anfordern |
| SHOW-PUBSET-ATTRIBUTES | Informationen über Pubset-Eigenschaften anfordern |
| SHOW-PUBSET-CACHE-ATTRIBUTES | Informationen über die Cache-Attribute eines Pubsets ausgeben |
| SHOW-PUBSET-DEFINITION-FILE | Informationen über die globalen und die Performance-Eigenschaften eines Volume-Sets ausgeben |

Tabelle 22: Kommandoübersicht zur Pubset-Verwaltung (Teil 2 von 3)

| Kommando | Bedeutung |
|---|---|
| SHOW-PUBSET-LOCKS | Informationen über Pubset-Locks ausgeben |
| SHOW-PUBSET-FILE-SERVICES | Die in einem SM-Pubset unterstützten Kombinationen von Dateieigenschaften anzeigen |
| SHOW-PUBSET-IMPORT-EXPORT | Information über den Bearbeitungsstand von Import-/ Exportaufträgen ausgeben |
| SHOW-PUBSET-OCCUPATION | Information über die Pubset-belegenden Jobs ausgeben |
| SHOW-PUBSET-PROCESSING | Information über die physikalische Pubset-Konfiguration ausgeben |
| SHOW-PUBSET-RESTRICTION | Information über die physikalische Pubset-Konfiguration ausgeben |
| SHOW-PUBSET-SPACE-ALLOCATION | Speicherplatzbelegung eines Pubsets ausgeben |
| SHOW-PUBSET-SPACE-DEFAULTS | Information über die Pubset-spezifischen Standardwerte für die Speicherzuweisung ausgeben |
| SHOW-PUBSET-SPACEPRO-HISTORY | SPACEPRO-Historie eines Pubsets anzeigen |
| SHOW-PUBSET-SPACEPRO-OPTIONS | SPACEPRO-Optionen eines Pubsets anzeigen |
| SHOW-SHARED-PUBSET | Übersicht über die Teilnehmer eines Shared-Pubset-Verbundes anfordern |
| SHOW-SPACE-SATURATION-LEVELS | Information über die Pubset-spezifischen Speichersättigungsstufen ausgeben |
| SHOW-SPACEPRO-PARAMETERS | SPACEPRO-Parameter anzeigen |
| SHOW-SPACEPRO-STATUS | Status des SPACEPRO-Monitors anzeigen |
| SHOW-XCS-PUBSET | Übersicht über die Teilnehmer eines XCS-Pubset-Verbundes anfordern |
| SHOW-XCS-OCCUPATION | Die momentan XCS-nutzenden TU-Tasks anzeigen |
| SIMULATE-SPACEPRO-EVENT | SPACEPRO-Konfiguration testen |
| START-PUBSET-CACHING | Cache-Puffer für einen Pubset aktivieren |
| START-SPACEPRO-MONITOR | SPACEPRO-Monitor für Pubset-Überwachung starten |
| STOP-PUBSET-CACHING | Cache-Puffer für einen Pubset deaktivieren |
| STOP-SPACEPRO-MONITOR | SPACEPRO-Monitor beenden |

Tabelle 22: Kommandoübersicht zur Pubset-Verwaltung (Teil 3 von 3)

## 9.1 Pubset-Konzept und SMS

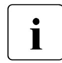
Als Dateiablageorte stehen dem BS2000-Benutzer externe Datenspeicher zur Verfügung. Darunter werden sowohl Magnetplatten, Net-Storage-Volumes wie auch Magnetbänder/MBKs verstanden.
Magnetplatten unterscheidet man in gemeinschaftliche und private Datenträger.
Mehrere gleichartige gemeinschaftliche Platten und ggf. der Net-Storage werden zu einem SF-Pubset oder einem Volume-Set zusammengefasst. Ein oder mehrere Volume-Sets bilden einen SM-Pubset.

#### Besonderheiten von Pubsets

Pubsets sind in sich abgeschlossene Behälter für Dateien und deren Metadaten. Sie können deshalb an unterschiedlichen BS2000-Systemen in Betrieb genommen werden.

Pubsets sind wegen ihrer Abgeschlossenheit weitestgehend voneinander unabhängige Objekte. Beim Ausfall eines Pubsets ist nur dieses davon betroffen.

SF-Pubsets werden immer als ganze Einheit, also vollständig, in oder außer Betrieb genommen (siehe Seite 360). Bei SM-Pubsets können einzelne Volume-Sets fehlen.



Ein Pubset, in dem Dateien auf Net-Storage katalogisiert sind, kann auch dann in Betrieb genommen werden, wenn der entsprechende Net-Storage nicht zur Verfügung steht. Es wird aber aus Verfügbarkeitsgründen empfohlen, den Net-Storage zuerst verfügbar zu machen und dann den Pubset in Betrieb zu nehmen, siehe Kapitel „Net-Storage-Verwaltung“ auf Seite 429.

#### Aufgaben der Pubset-Verwaltung

Grundidee des Pubset-Konzepts ist die klare Trennung der Aufgaben des Benutzers von den Aufgaben der Pubset-Verwaltung. Letztere werden durch den Systembetreuer oder durch automatische Systemroutinen übernommen.

1. Bereitstellen bzw. Entziehen des Betriebsmittels Pubset im Rahmen des DVS:
   Beim Bereitstellen eines Pubsets werden alle notwendigen Datenstrukturen initialisiert und die entsprechenden Betriebsmittel (Speicherplatz, Zugriffsrechte) angefordert. Beim Entziehen eines Pubsets werden die Strukturen abgebaut und die angeforderten Betriebsmittel freigegeben. Realisiert werden diese Aufgaben durch die Funktionen

   – „Pubset importieren“: eingeleitet durch das Kommando IMPORT-PUBSET oder automatisch während der Systemeinleitung (Startup)

   – „Pubset exportieren“: eingeleitet durch das Kommando EXPORT-PUBSET oder automatisch während der Systembeendigung (Shutdown)

2. Festlegen des Ablageortes der Datei:
   Hierbei soll dem Benutzer die Entscheidung abgenommen werden, auf welchem Datenträger und ggf. auf welchem Ablageort innerhalb des Datenträgers seine Dateien liegen sollen. Folgende Faktoren spielen bei der Platzallokierung eine wesentliche Rolle:

   – Für welche Platten besteht eine Allokierungsberechtigung?
   – Soll die Datei zusammen mit bestimmten anderen Dateien einer Anwendung auf einer bestimmten Platte liegen?
   – Ist auf einer Platte ausreichend Platz vorhanden?
   – Decken sich die Hardware-Eigenschaften der Platte (z.B. Performance, Verfügbarkeit, usw.) mit den Anforderungen für die Datei?

   Kommen die Dateien auf Pubsets zur Ablage, braucht sich der Benutzer um diese Faktoren nicht zu kümmern; diese Aufgabe wird ihm vom DVS abgenommen.

   Darüber hinaus kann der Benutzer seine Dateien bzgl. Performance, Verfügbarkeit und Strukturmerkmale entsprechend attributieren (Kommando CREATE-FILE oder MODIFY-FILE-ATTRIBUTES). Anhand dieser Attribute können dann die Selektionsroutinen des DVS den am besten geeigneten Dateiablageort innerhalb des Pubset ermitteln. Es ist also nur die Kenntnis des Pubset-Namens erforderlich. Wird zusätzlich die Standard-Katalogkennung – jedem Benutzer ist im Benutzerkatalog standardmäßig ein Pubset zugeordnet, auf dem all seine Dateien zur Ablage kommen, falls keine Angabe bzg. des Pubset-Namen gemacht wird – genutzt, bleibt der Dateiablageort für den Benutzer vollständig transparent.

3. Allokierungsanforderung:
   Die Größe einer Allokierungsanforderung wird in geräte-unabhängigen Einheiten angegeben. Der Benutzer muss lediglich darauf achten, dass durch die Anforderung sein entsprechendes Kontingent nicht überschritten wird (siehe Abschnitt „Benutzer-Kontingente und Ressourcen-Schutz“ auf Seite 355). Das Plattenformat zur Berechnung von geräte-spezifischen Allokierungseinheiten bleibt dem Benutzer somit verborgen.

4. Dynamische Erweiterbarkeit:
   Plattenspeicherplatz kann während der Dateiverarbeitung dynamisch im Rahmen der Benutzerkontingente nachgefordert werden, d.h., die endgültige Größe einer Datei muss zum Zeitpunkt des Einrichtens nicht bekannt sein.

   Die Aufgabe des Systembetreuers ist es, dem Benutzer die erforderliche Datenträgerperipherie verfügbar zu machen, aus der die system-spezifischen Selektionsroutinen die geeigneten Dateiablageorte auswählen, um für diese Peripherie geeignete Reorganisationsmaßnahmen anstoßen bzw. durchführen zu können. Für eine effiziente Verwaltung werden im Pubset-Konzept geeignete Hilfsmittel, z.B. die Funktionen der Pubset-Rekonfiguration (siehe Seite 367) zur Verfügung gestellt.

Die Eigenschaften eines Volume-Sets haben Auswirkungen auf seine Nutzbarkeit als Dateiablageort:

- SM-Pubsets sind Pubsets, die aus verschiedenartigen Volume-Sets bestehen, und somit ein Spektrum von Eigenschaften aufweisen, die Einfluss auf die Qualität des Pubsets als Dateiablageort haben. Beim Einrichten einer neuen Datei auf einem solchen Pubset entscheidet das DVS anhand der logischen Dateieigenschaften, welches Volume-Set sich am besten als Ablageort für die Datei eignet.
  Bild 16 auf Seite 325 zeigt den prinzipiellen Aufbau eines SM-Pubsets.
- SF-Pubsets bestehen aus Volumes mit gleichen Eigenschaften und weisen deshalb auch nur genau diese Eigenschaften auf.
  Bild 15 auf Seite 320 zeigt den prinzipiellen Aufbau eines SF-Pubsets.

### 9.1.1 Pubset-Arten

**Home-Pubset**

In jedem BS2000/OSD-System muss mindestens ein Pubset existieren, der system-spezifische Daten und Dateien beinhaltet und deshalb während der gesamten BS2000-Session verfügbar sein muss. Dieser spezielle Pubset wird als Home-Pubset bezeichnet. Er kann nicht mit Net-Storage verbunden werden.

Zur Identifikation des laufenden Systems wird die Systemkennung (Sysid) des Home-Pubsets verwendet. Sie geht in die Plattenbelegung und in die Pfadgruppen-Id ein. Die Pfadgruppen-Id dient Controllern zur Unterscheidung von I/Os verschiedener Systeme. Aus diesen Gründen ist es notwendig, dass bei parallel betriebenen Home-Pubsets die Systemkennung eindeutig vergeben wird. Bei der PUB-Notation (siehe Seite 316) ist dies von selbst gegeben, sofern die Pubset-Kennungen eindeutig gewählt werden. Bei Pubsets mit Punkt-Notation (siehe Seite 317), die als Home-Pubsets verwendet werden sollen, muss explizit auf eine eindeutige Systemkennung geachtet werden. Zur Sysid siehe Seite 398.

Das Importieren des Home-Pubsets wird während der Systemeinleitung durch BS2000/OSD selbst veranlasst. Das Exportieren des Home-Pubsets geschieht während der Systembeendigung (bei Abarbeitung des Kommandos SHUTDOWN).
Es ist möglich, auf dem Home-Pubset Benutzerdateien und Jobvariablen zu katalogisieren. Dies muss allerdings von der Systembetreuung zugelassen werden (ADD-USER- oder MODIFY-USER-Kommando).

### Benutzer-Pubset (Daten-Pubset)

Diese Pubsets dienen dem DVS ausschließlich als Dateiablageorte. Sie können auch mit Net-Storage verbunden werden. Sie enthalten nur Benutzerdateien oder -Jobvariablen. Die Verfügbarkeit dieser Pubsets kann mit den Kommandos IMPORT-PUBSET und EXPORT-PUBSET jederzeit während des Systemlaufs von der Systembetreuung gesteuert werden. Ein Benutzer kann auf mehreren Pubsets zugriffsberechtigt sein.

Jedem Benutzer wird ein Standard-Pubset (Default-Pubset) zugewiesen. Auf diesem Pubset kann er (ohne Angabe der Katalogkennung) Dateien und Jobvariablen erzeugen, verarbeiten und löschen.
Für die Benutzerkennung TSOS ist unabhängig von der DEFAULT-PUBSET-Angabe im ADD-USER- bzw. MODIFY-USER-ATTRIBUTES-Kommando der Standard-Pubset immer der Home-Pubset.

### Standby-Pubset

Dieser Pubset enthält dieselben Dateien wie der Original-Pubset und kann bei einem Ausfall des Original-Pubsets dessen Funktionen übernehmen. Abhängig von Art des Original-Pubsets (Home- oder Daten-Pubset) ist der Ersatz-Pubset ein **Standby-Home-Pubset** oder ein **Standby-Daten-Pubset**.

- Bei Ausfall des Home-Pubsets ist das System nicht mehr ablauffähig und muss mit dem Standby-Home-Pubset neu hochgefahren werden. Anschließend kann der ausgefallene Home-Pubset rekonstruiert werden.
- Bei Ausfall eines Daten-Pubsets sind Anwendungen, die auf diesen Pubset zugreifen, u. U. nicht mehr ablauffähig. Den Nutzern dieses Pubsets kann durch Importieren des Standby-Daten-Pubsets die Wiederaufnahme der Verarbeitungsläufe auf einem bestimmten Konsistenzpunkt ermöglicht werden. Der ausgefallene Daten-Pubset kann ggf. durch die Systembetreuung rekonstruiert werden.

Das Erstellen und die Pflege von Standby-Pubsets ist Aufgabe der Systembetreuung; es existieren keine Automatismen durch das Betriebssystem. Der Home-Pubset bzw. ein Daten-Pubset können jedoch mit Spiegelplattentechnik auf Hardware-Basis (RAID1) oder auf Software-Basis (DRV) betrieben werden.

Plattenspeichersysteme bieten bei Einsatz des Software-Produkts SHC-OSD (Storage Host Component für BS2000/OSD) mit ihren Replikationsfunktionen Schnittstellen zur Online-Erstellung von Standby-Pubsets. Siehe den Abschnitt „Erhöhte Datensicherheit durch Standby-Pubsets“ auf Seite 404. Weitere Einzelheiten zu den Funktionen enthält das Handbuch „SHC-OSD“ [50].

### Paging-Pubset

Die Pubsets, die die im Systemlauf zu verwendenden Paging-Dateien enthalten, werden als Paging-Pubsets bezeichnet. Sie werden für die gesamte BS2000-Session benötigt, können jedoch davon unabhängig – aber nur vom gleichen System – für DVS-Nutzung beliebig in und außer Betrieb genommen werden.

Paging-Pubsets werden vom DVS wie die Pubsets behandelt, auf denen keine Paging-Dateien angelegt sind: der Import-Auftrag für einen Daten-Pubset mit Paging-Datei muss explizit erfolgen (Kommando IMPORT-PUBSET). Ein automatisches Importieren bei der Systemeinleitung findet für die Paging-Pubsets nicht statt. Um die benötigten Pubsets sofort nach der Systemeinleitung zur Verfügung zu haben, sollte deren Import über den Anstoß einer RUN-Datei realisiert werden.
Die Paging-Pubsets sind im Katalogverzeichnis MRSCAT als solche gekennzeichnet und können dort nicht gelöscht werden.
Mit dem Kommando SHOW-MASTER-CATALOG-ENTRY CATALOG-ID=*ALL,SELECT=*PAGING lassen sich alle Paging-Pubsets des Systems anzeigen.

### Pubsets im Rechnerverbund – Shared-Pubsets

Bei Einsatz des Produkts HIPLEX MSCF und einer entsprechenden Hardware-Konfiguration ist der gleichzeitige und gemeinsame Zugriff über max. 16 Systeme (BS2000/OSD native oder Gastsysteme unter VM2000) hinweg auf einen Pubset möglich. Dieser mehrbenutzbare Pubset wird „Shared-Pubset“ genannt.

Das Kommando SHOW-SHARED-PUBSET gibt die wichtigsten Kenndaten der Sharer eines oder aller Shared-Pubsets an, die diesen Pubset seit dem letzten Master-Import importiert haben.

Mit HIPLEX MSCF steht neben dem Shared-Pubset-Verbund eine erweiterte Verbundfunktionalität zur Verfügung: der XCS-Verbund (Cross-Coupled-System).
Der XCS-Verbund bietet eine engere Koordination der beteiligten Systeme als ein Shared-Pubset-Verbund. Ein „XCS-Pubset“ dient als zentraler Ablageort für verbundweit benötigte Daten. XCS-Pubsets werden automatisch durch das System importiert.

Das gesamte Konzept des Shared-Pubset (Hardware-Konfiguration, Verwaltung der Pubsets, Datenzugriffe, Watch-Dog-Mechanismus) und des XCS-Verbunds (Funktionalität, Generierung und Betrieb) ist ausführlich im Handbuch „HIPLEX MSCF“ [33] beschrieben.

Zu Aufgaben der Systembetreuung in Verbindung mit einem Shared-Pubset- oder XCS-Verbund siehe Abschnitt „Shared-Pubsets“ auf Seite 398.

### Pubsets mit Volumes und Dateien ≥ 32 GB

Es werden Volumes und Dateien mit einer Größe von mehr als 32 GB unterstützt. Volumes und Dateien, deren Größe 32 GB überschreitet, werden auch als „große Volumes“ bzw. „große Dateien“ oder gemeinsam als „große Objekte“ bezeichnet. Die max. mögliche Größe beträgt 4 TB.

Zur Unterstützung großer Objekte werden zwei Pubset-Typen angeboten:

- Pubsets mit großen Volumes ohne große Dateien
- Pubsets mit großen Volumes und großen Dateien

Damit ist es möglich, große Volumes und große Dateien gezielt stufenweise einzuführen. Die Einführung großer Volumes ist weitestgehend transparent für existierende Programme. Bei der Einführung großer Dateien sind u.U. Anpassungen vorzunehmen.

Bestehende Pubsets können mit dem Kommando SET-PUBSET-ATTRIBUTES zu großen Pubsets ohne und mit Unterstützung großer Dateien hochgestuft werden.

```
/SET-PUBSET-ATTRIBUTES ...,LARGE-VOLUMES=*UNCHANGED/*ALLOWED(
                           LARGE-FILES=*UNCHANGED/*ALLOWED)
```

Dabei sind folgende Hochstufungen möglich:

| | | |
|---|---|---|
| Kleiner Pubset | zu | Großer Pubset ohne große Dateien |
| Kleiner Pubset | zu | Großer Pubset mit großen Dateien |
| Großer Pubset ohne große Dateien | zu | Großer Pubset mit großen Dateien |

Beim Import des Pubsets werden diese Attribute in den MRSCAT übernommen und durch die entsprechenden Auskunftsfunktionen angezeigt:

- Kommando SHOW-MASTER-CATALOG-ENTRY
- Programmschnittstelle STAMCE

Zum Einrichten und Erweitern von (großen) Pubsets mit SIR siehe das Handbuch „Dienstprogramme“ [15].



Ein HOME-Pubset darf große Platten enthalten, jedoch keine großen Dateien. Ein Startup von einem Pubset mit `LARGE-FILES-ALLOWED=*YES` wird abgebrochen.

Die Erlaubnis zu großen Volumes und Dateien kann einem Pubset nicht wieder entzogen werden. Ein Pubset, der große Volumes und Dateien zulässt, ist nicht kompatibel zu BS2000/OSD < V5.0A.

Mit den Kommandos SHOW-PUBSET-ATTRIBUTES und SHOW-MASTER-CATALOG-ENTRY können alle Pubsets mit den Attributen „große Volumes“ und „große Dateien“ angezeigt werden.

### 9.1.2 VSN und Pubset-Adressierung

Alle Datenträger werden in BS2000/OSD durch einen Namen identifiziert, der bis zu sechs Zeichen lang sein darf und VSN (Volume Serial Number) genannt wird. Der gültige Zeichenvorrat besteht aus den alphanumerischen Zeichen A...Z und 0...9 sowie den Sonderzeichen @, # und $. Um zwischen privaten und gemeinschaftlichen Platten allein auf Basis der VSN-Syntax unterscheiden zu können, unterliegen die VSN der gemeinschaftlichen Platten einer Konvention: Sie müssen entweder mit der Zeichenkette PUB beginnen oder an der dritten, vierten oder fünften Position einen Punkt enthalten. Diese VSN-Syntax darf für private Platten nicht angewandt werden.

Die Namen von Platten, die einem Volume-Set/SF-Pubset zugeordnet sind, müssen syntaktisch zum Namen des Volume-Sets/SF-Pubsets passen. Dabei ist zwischen einstelligen und mehrstelligen Volume-Set- bzw. Pubset-Namen zu unterscheiden:

- für einstellige Namen gibt es die PUB-Notation
- für zwei- bis vierstellige Namen gibt es die Punkt-Notation
- für Net-Storage-Volumes gibt es eine spezielle Notation, siehe Seite 432

#### VSN in PUB-Notation

Diese Art der Pubset-Adressierung hat ihren Namen aus der Verwendung der Zeichenkette „PUB“ als festen, unveränderlichen Bestandteil der VSN. Dabei weist „PUB“ (public) darauf hin, dass es sich um gemeinschaftliche Platten handelt.

Eine VSN in PUB-Notation besteht immer aus 6 Zeichen und hat das Format **PUBpxx.** Dabei ist:

**PUB** unveränderlicher Bestandteil zur Unterscheidung von privaten Platten (3 Zeichen „PUB“) = Typbezeichner

**p** Katalogkennung (Catid), (1 Zeichen; A..Z, 0..9)

**xx** Nummer innerhalb eines Pubsets/Volume-Sets, (2 Zeichen; 00..31) = Folgenummer

Mit der einstelligen Katalogkennung können maximal 36 Pubsets bzw. Volume-Sets adressiert werden, die aus bis zu 32 Platten bestehen können.

Beispiele: PUBA00, PUBA25, PUB502

### VSN in Punkt-Notation

Der Name dieser Notation ist durch die Verwendung eines Punktes als Trennungszeichen zwischen der Katalogkennung und der Folgenummer im Pubset bzw. Volume-Set entstanden. Auch die Punkt-Notation bezeichnet immer gemeinschaftliche Platten.

Eine VSN in Punkt-Notation besteht immer aus 6 Zeichen und hat das Format **pp[pp].[xy]z.** Dabei ist:

**pp[pp]** Katalogkennung (Catid), (2-4 Zeichen; je A..Z, 0..9); Präfix „PUB“ ist nicht erlaubt

**.** Punkt (1 Zeichen) = Typbezeichner

**[xy]z** Nummer innerhalb eines Pubsets/Volume-Sets (1-3 Zeichen) = Folgenummer

Ein Pubset bzw. Volume-Set in Punkt-Notation kann aus bis zu 255 Platten bestehen.

Beispiele: AA.001, AB.309, XYZ.23, OTTO.0, J19P.8

Die Katalogkennung bei SF-Pubsets korrespondiert direkt mit dem Namensteil „Katalogkennung“ in der VSN. Bei SM-Pubsets ist die Katalogkennung verschieden von allen Volume-Set-Namen dieses SM-Pubsets, und damit auch verschieden vom VSN-Namensteil „Katalogkennung“ aller Platten des Pubsets.

### Maximale Anzahl von Platten pro Pubset/Volume-Set in Punkt-Notation

Die maximale Anzahl der Platten pro Pubset/Volume-Set hängt von der Allokierungseinheit AU (ALLOCATION-UNIT) der Plattentypen und von der Länge der Folgenummer ab.

| AU | Wertebereich der Folgenummer | | | maximale Anzahl adressierbarer Platten |
|---|---|---|---|---|
| | x | y | z | |
| 6 KB | 0 | 0 | A..V, 0..9 | 32 |
| 8 KB | A..Z, 0..9 | A..Z, 0..9 | A..Z, 0..9 | 36 / 255 [1] |
| 64 KB | A..Z, 0..9 | A..Z, 0..9 | A..Z, 0..9 | 36 / 255 [1] |

[1] bei einstelliger Folgenummer können nur maximal 36 Platten pro Pubset adressiert werden

### Konvertierung oder Umbenennung des VSN-Formats

Mit dem Dienstprogramm PVSREN (siehe Handbuch „Dienstprogramme“ [15]) können Pubset-Notationen konvertiert oder Pubset- bzw. Volume-Set-Bezeichnungen innerhalb einer Notationsart umbenannt werden.

Eine Konvertierung von PUB-Notation in Punkt-Notation ist immer möglich.
Eine Konvertierung von Punkt-Notation in PUB-Notation ist nur dann möglich, wenn die Anzahl der Volumes des betroffenen Pubsets bzw. Volume-Sets nicht größer als 32 ist.

### Doppelpunkt-Notation

Die Doppelpunkt-Notation dient dazu, Spiegelplatten eindeutig zu bezeichnen, nachdem sie von der Original-Unit abgetrennt wurden (z.B. zur Durchführung einer Sicherung mit HSMS). Spiegelplatten werden mit den Replikationsfunktionen externer Plattenspeichersysteme erstellt.

Dazu wird in einer VSN mit Punkt-Notation der Punkt in einen Doppelpunkt bzw. bei PUB-Notation die Zeichenfolge „PUB“ nach „P:B“ geändert.

*Beispiele*

| | | | **Doppelpunkt-Notation** |
|---|---|---|---|
| **Punkt-Notation:** | ABC.04 | wird zu | ABC:04 |
| | XY.123 | wird zu | XY:123 |
| **PUB-Notation:** | PUB023 | wird zu | P:B023 |
| | PUBX88 | wird zu | P:BX88 |

Bei der Erstellung eines Spiegel-Pubsets mit SHC-OSD kann diese Umbenennung implizit erfolgen. Es können jedoch nur SF-Pubsets implizit umbenannt werden.

### Sonderfall Snapset-Notation

Für Snap-Units, die bei der Erzeugung eines Snapsets verwendet werden, wird zur eindeutigen Benennung der VSNs die Snapset-Identifikation in Kleinbuchstaben so eingesetzt, dass einerseits aus der Snapset-Platte die zugehörige Pubset-Platte abgeleitet werden kann und andererseits die VSN der Snapset-Platte außerhalb des Namensraums der VSNs für Privatplatten und Pubsets liegt. (Details siehe „Snapset-Identifikation“ auf Seite 543).

### Pubset-Adressierung

Pubsets werden durch Namen identifiziert. Innerhalb eines Rechnerverbundes müssen die Pubset-Namen eindeutig sein.

Die Adressierung der auf einem Pubset katalogisierten Objekte erfolgt über die Katalogkennung (Catid) durch die vollständigen Pfadnamen.

Die Catid ist identisch mit dem Namen des korrespondierenden Pubsets.
Die Catid eines SM-Pubsets muss verschieden von allen Volume-Set-Namen dieses SM-Pubsets sein.

Die Adressierung von Dateien auf einem bestimmten Volume-Set eines SM-Pubsets über den Namen dieses Volume-Sets wird nicht unterstützt.

## 9.2 SF-Pubsets

Ein Single-Feature-Pubset (SF-Pubset) besteht aus einer oder mehreren homogenen Platten, die in den wesentlichen Eigenschaften (Plattenformat, Allokierungseinheit) übereinstimmen müssen. Zur Pubres siehe auch Seite 349.

*Beispiel*

Der SF-Pubset mit der Katalogkennung ABC besteht aus drei Platten, die folgende VSNs (in Punkt-Notation) haben: ABC.00, ABC.01 und ABC.02.

Existiert eine Datei mit dem Dateinamen „MEINE.LISTE“ unter der Benutzerkennung „ALLERLEI“ auf einer der Platten des SF-Pubsets ABC, so lautet ihr Pfadname: „:ABC:$ALLERLEI.MEINE.LISTE“

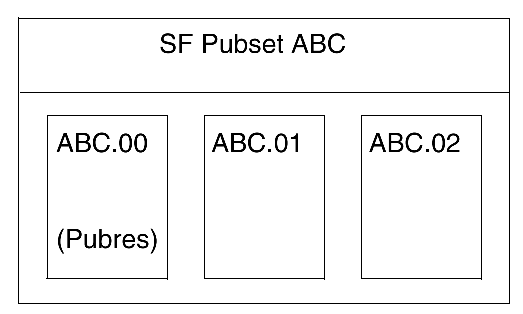



Bild 15: Datenträgerkonfiguration eines SF-Pubsets (Beispiel)

### Service-Angebot

Ein SF-Pubset repräsentiert ein begrenztes, in sich homogenes Service-Angebot. Ändern sich die Service-Anforderungen einer Datei, muss die Datei vom Benutzer auf einen anderen Pubset verlagert werden.

### Rekonfiguration

SF-Pubsets können durch Hinzunahme leerer Platten vergrößert werden. Die Platteneigenschaften müssen dabei den Pubset-Eigenschaften entsprechen (siehe Abschnitt „Einrichten und Erweitern eines SF-Pubsets“ auf Seite 321).
Durch Herausnehmen leerer Platten kann ein SF-Pubset verkleinert werden. Als Hilfsmittel für das Leeren von Platten wird seitens des Systems das Setzen von Allokierungseinschränkungen zur Verfügung gestellt.
Die Rekonfiguration eines SF-Pubsets ist mit SIR (nur Hinzunahme von Platten) oder dem Kommando MODIFY-PUBSET-PROCESSING möglich.

Die Anzeige der aktuell zu einem Pubset gehörenden Platten erfolgt mit dem Kommando SHOW-DEVICE-CONFIGURATION UNIT=*PUBSET-DEVICES(...).
Mit dem Kommando SHOW-PUBSET-PROCESSING kann die aktuelle Pubset-Konfiguration des Systems ausgegeben werden.

#### Ausfalleinheit

Bei Ausfall nur eines einzelnen Volumes des SF-Pubsets gilt der gesamte Pubset als ausgefallen.

#### Systemfunktionalität

SF-Pubsets sind als Home- oder Paging-Pubsets geeignet.

#### Einrichten und Erweitern eines SF-Pubsets

SF-Pubsets werden mit dem Dienstprogramm SIR eingerichtet und erweitert.

Informationen zum Einrichten von SF-Pubsets und Home-Pubsets sowie zum Erweitern von SF-Pubsets mit SIR finden Sie bei der Beschreibung des Dienstprogramms SIR im Handbuch „Dienstprogramme“ [15]. Dort finden Sie auch Hinweise zum Einrichten und Erweitern von Pubsets mit Volumes und Dateien ≥ 32 GB.

Alternativ zum Erweitern von Pubsets mit SIR besteht die Möglichkeit, die neuen Platten mittels VOLIN mit der entsprechenden VSN und dem entsprechenden Format zu initialisieren. Beim nächsten Import des Pubsets mit IMPORT-PUBSET (nicht beim impliziten Import während des Startup) fragt das System, ob die neuen Platten zum Pubset hinzugenommen werden sollen. Das Dienstprogramm VOLIN ist ebenfalls im Handbuch „Dienstprogramme“ [15] beschrieben.

#### SF-Pubset und Net-Storage

Zum Verbinden von SF-Pubsets mit Net-Storage siehe Kapitel „Net-Storage-Verwaltung“ auf Seite 429.

### Ersetzen und Entfernen von SF-Pubsets mit gleichem Namen

Der Ersatz eines Pubsets durch einen gleichnamigen Pubset geschieht wie folgt:

1. eventuell benötigte Informationen des alten Pubsets sichern
2. den alten Pubset exportieren
3. die Platten des alten Pubsets mit dem Kommando DETACH-DEVICE (siehe Handbuch „Kommandos“ [27]) oder physikalisch wegschalten bzw. umbenennen
4. die Platten des neuen Pubsets mit SIR generieren
5. den neuen Pubset mit IMPORT-PUBSET ..,ACTUAL-JOIN=*FIRST importieren
6. eventuell gesicherte Dateien einspielen

Der Ersatz eines Pubsets durch einen gleichnamigen Pubset aus einem anderen System geschieht in folgenden Schritten:

1. eventuell benötigte Informationen des alten Pubsets sichern
2. den alten Pubset exportieren
3. die Platten des alten Pubsets mit dem Kommando DETACH-DEVICE oder physikalisch wegschalten bzw. umbenennen
4. den MRSCAT-Eintrag des Pubsets (enthält die MN der Pubres) löschen und neu einrichten
5. den neuen Pubset im anderen System exportieren
6. die Platten des neuen Pubsets mit dem Kommando ATTACH-DEVICE zuschalten
7. den neuen Pubset importieren

Wenn der neue Pubset als Home- und Paging-Pubset verwendet werden soll, dann müssen die Platten des alten gleichnamigen Pubsets vor der nächsten Systemeinleitung umbenannt oder physikalisch weggeschaltet werden, damit die Menge der physikalisch verfügbaren Pubsets eindeutig bleibt.

Falls dies versehentlich nicht passiert, muss der Benutzer bei Systemeinleitung mittels geführtem Dialog die benötigten Platten sorgfältig auswählen.

## 9.3 SM-Pubsets

System-Managed-Pubsets (SM-Pubsets) zeigen im Vergleich zu SF-Pubsets einen komplexeren Aufbau. Ein SM-Pubset besteht aus einem oder mehreren **Volume-Sets**, die wie bei einem SF-Pubset eine Zusammenfassung von mehreren homogenen Platten sind; die Homogenität bezieht sich auch hier auf bestimmte physikalische Eigenschaften wie z.B. Plattenformat und Allokierungseinheit.

Ein Volume-Set kann nur innerhalb des SM-Pubsets betrieben werden. Er hat folgende Eigenschaften:

- Er wird durch eine Volume-Set-Kennung identifiziert, die analog zur Katalogkennung (catid) für SF-Pubsets den Hauptbestandteil der VSN aller zum Volume-Set gehörenden Platten bildet.
- Alle zugehörigen Platten haben ein einheitliches Plattenformat (K, NK2, NK4) und eine einheitliche Allokierungseinheit.
- Einem Volume-Set kann maximal ein Cache-Bereich zugeschaltet werden.
- Jeder Volume-Set bildet einen Subbehälter für komplette Dateien, d.h. einzelne Dateien können sich nicht über mehrere Volume-Sets erstrecken.
- Jeder Volume-Set besitzt eine sog. **Volres**, in der i.W. die physikalische Konfiguration des Volume-Sets hinterlegt ist (siehe auch Seite 349) .

Ein Volume-Set kann folgende Betriebszustände einnehmen:

| | |
|---|---|
| DEFECT | der Volume-Set ist defekt |
| DEFINED-ONLY | der Volume-Set ist nur definiert, aber nicht zugreifbar |
| IN-HOLD | der Volume-Set ist temporär nicht in Betrieb |
| NORMAL-USE | der Volume-Set ist in Betrieb und damit zugreifbar |
| CONTROL-VOLSET | der Volume-Set ist der Control-Volume-Set |

Der **Control-Volume-Set** dient zur Ablage aller pubset-globalen Metadaten des SM-Pubsets und enthält:

- die Pubset-Konfigurationsdatei (Pubset Configuration File)
  (diese Datei enthält eine Liste aller Volume-Sets, die zum SM-Pubset gehören)
- Spezialkataloge für Jobvariablen, migrierte Dateien, No-Space-Dateien und Dateien auf privaten Datenträgern
- den Benutzerkatalog
- den Guards-Katalog

Als „Kern“ des SM-Pubsets sollte er vor allem für „große“ SM-Pubsets, die aus vielen Volume-Sets bestehen, besonders ausfallsicher sein. Dazu kann er hardwaremäßig gespiegelt werden (z.B. mit RAID1 oder RAID1/0). Eine Spiegelung ist auch auf Volume-Ebene (mit dem Software-Produkt DRV) möglich.

Bild 16 auf Seite 325 zeigt den prinzipiellen Aufbau eines SM-Pubsets. Obligatorischer Bestandteil eines SM-Pubsets ist seine Verarbeitungsebene. Sie besteht aus einem oder mehreren Volume-Sets. Unter den Volume-Sets nimmt der Control-Volume-Set eine Sonderrolle ein.

SF- und SM-Pubsets können mit Hilfe des Produkts HSMS mit Hintergrundebenen ausgestattet werden. Diese bilden zusammen mit der Verarbeitungsebene bzgl. Zugriffszeit, Verfügbarkeit und Kosten eine dreistufige Speicherhierarchie. Die Position innerhalb der Hierarchie wird durch die Bezeichnungen S0-Ebene (für die Verarbeitungsebene), S1-Ebene (für die online verfügbare Hintergrundebene) und S2-Ebene (für die aus offline zugreifbaren Datenträgern gebildete Hintergrundebene) verdeutlicht. Wenn Dateien längere Zeit nicht bearbeitet werden, ist es empfehlenswert, sie auf eine kostengünstigere Hintergrundebene zu verdrängen.
Bei SM-Pubsets wird die S1-Ebene durch einen zum Pubset gehörenden Volume-Set realisiert, der ausschließlich für diesen Zweck reserviert sein muss. Als Datenträger für die S2-Ebene dient ein aus Magnetbändern (oder MBK) bestehender Band-Pool. Die Hintergrundebenen sind einem SM-Pubset exklusiv zugeordnet und können – im Unterschied zu den Hintergrundebenen von SF-Pubsets – nicht von mehreren Pubsets genutzt werden. Die für die Benutzung der Hintergrundebenen benötigten Metadaten, wie z.B. das Directory des Migrations-Archivs, befinden sich in dem SM-Pubset selbst und liegen dort auf dem Control-Volume-Set.

Der Benutzer muss den Aufbau des SM-Pubsets nicht kennen. Insbesondere ist die Adressierung einer Datei unabhängig von ihrer Lage innerhalb des Pubsets.

*Beispiel zum* *Bild 16 auf Seite 325*

> Der SM-Pubset mit der Katalogkennung XYZ besteht in der Verarbeitungsebene aus drei Volume-Sets.
> Existiert eine Datei mit dem Dateinamen „LST.TELEFON“ unter der Benutzerkennung „EINERLEI“ auf einer der Platten, die zu irgendeinem der Volume-Sets des SM-Pubsets XYZ zusammengefasst sind, so lautet ihr Pfadname:
> „:XYZ:$EINERLEI.LST.TELEFON“, d.h. für den Benutzer ist es i.A. nicht relevant, wie der SM-Pubset intern aufgebaut ist und wo genau seine Datei vom System abgelegt wurde.

SM-Pubsets, die aus genau einem Volume-Set bestehen, sind funktionell SF-Pubsets sehr ähnlich. Sie unterscheiden sich jedoch intern durch den Aufbau der Metadatenstruktur.

**SM-Pubset XYZ**

S0-Ebene (Verarbeitungsebene)

Volume-Set AB
Control-Volume-Set

pubset-globale
Metadaten

AB.000
Volres

AB.001

Volume-Set OTTO

OTTO.0
Volres

Volume-Set A

PUBA00
Volres

PUBA02

PUBA21

S1-Ebene (Hintergrundebene)

Volume-Set ARC

für Datei-Migration

ARC.00
Volres

ARC.01

S2-Ebene (Hintergrundebene)

Tape-Pool

für Datei-Migration

Backup-Konfiguration

Tape-Pool

Bild 16: SM-Pubset-Struktur mit HSMS-Konfiguration

### Service-Angebot

Jedes Volume-Set repräsentiert anhand seiner Eigenschaften einen speziellen Service-Typ innerhalb des SM-Pubsets. Volume-Sets mit gleichen Eigenschaften werden dem gleichen Service-Typ zugeordnet. Der Service-Typ wird bei der Auswahl eines Volume-Sets berücksichtigt. Grundlage für die Auswahl sind die vom Benutzer durch Dateiattribute angeforderten Services (Performance, Verfügbarkeit, Nutzung spezieller Anwendungen). Beim Neuanlegen einer Datei entscheidet das System anhand einer Volume-Set-Selektionsroutine (siehe Seite 331), welches Volume-Set sich im Hinblick auf die Benutzeranforderung am besten als Dateiablageort eignet.

Ändern sich die Eigenschaften einer Datei während ihrer Lebensdauer durch Eingriffe des Benutzers oder erscheint aus Gründen der optimalen Ressourcen-Nutzung eine Umverteilung von Dateien als wünschenswert, können die Dateien innerhalb des SM-Pubset auf ein als Ablageort besser geeignetes Volume-Set verlagert werden, ohne die Adressierung zu ändern, d.h. die Verlagerung geschieht nicht sichtbar für den Benutzer.

Mit dem Kommando SHOW-PUBSET-FILE-SERVICES kann das bestehende Service-Angebot eines SM-Pubsets ausgegeben werden.

### Rekonfiguration

Für die Rekonfiguration eines SM-Pubsets im laufenden Betrieb stehen umfangreiche Funktionen zur Verfügung. Ein SM-Pubset kann während der Betriebsphase um leere Volume-Sets erweitert oder verkleinert werden. Leere Volumes können zu einzelnen Volume-Sets hinzugefügt oder aus ihnen entfernt werden.
Zum Leeren von Volume-Sets und Volumes werden Hilfsfunktionen angeboten.
Dateien können – für den Benutzer transparent – innerhalb des Pubset verlagert werden.

Die statische Pubset-Konfiguration (Volume-Set-Konfiguration und Eigenschaftsprofile der Volume-Sets) kann mit dem Kommando MODIFY-PUBSET-DEFINITION-FILE (EDIT-PUBSET-DEFINITION-FILE) geändert werden. Die Änderungen werden sowohl in der Pubset-Konfigurationsdatei als auch im MRSCAT vollzogen.
Die dynamische Pubset-Konfiguration (Volume-Set-Konfiguration des SM-Pubsets und Plattenkonfiguration von Volume-Sets) wird mit dem Kommando MODIFY-PUBSET-PROCESSING geändert.

Zur Rekonfiguration mit SIR siehe Abschnitt „Einrichten und Erweitern von Volume-Sets und SM-Pubsets“ auf Seite 327. Auf Kommandoebene können mit MODIFY-PUBSET-PROCESSING einzelne Volume-Sets zu einem SM-Pubset hinzugefügt oder aus ihm entfernt werden.

Die Anzeige der aktuell zu einem Pubsest gehörenden Platten erfolgt mit dem Kommando SHOW-DEVICE-CONFIGURATION UNIT=*PUBSET-DEVICES(...).
Mit dem Kommando SHOW-PUBSET-PROCESSING kann die aktuelle Pubset-Konfiguration des Systems ausgegeben werden.

### Ausfalleinheit

Bei Ausfall eines einzelnen Volumes eines SM-Pubsets beschränkt sich der Schaden auf den betroffenen Volume-Set. Das hat zur Folge, dass lediglich alle auf dem betroffenen Volume-Set liegenden Dateien zerstört sind, der restliche Teil des Pubsets ist nicht betroffen und kann ohne Unterbrechung weiterbenutzt werden. Der defekte Volume-Set kann relativ problemlos aus dem Pubset entfernt werden, wobei gleichzeitig eine Liste aller betroffenen Dateien erstellt wird. Diese Liste (abgelegt in der Datei $TSOS.SYS.PUBSET.DEFECT.<volume-set-id>.<datum.uhrzeit>) dient als Input für einen anschließend möglichen Restore-Auftrag für die defekten Dateien. Diese Eigenschaften des SM-Pubsets erlauben es – im Gegensatz zu SF-Pubsets – sehr große Pubsets zu bilden, ohne das Ausfallrisiko für den gesamten Pubset zu vergrößern.

### Systemfunktionalität

SM-Pubsets können nicht als Home-Pubsets verwendet werden.

Paging-Dateien können auf SM-Pubsets eingerichtet werden. Bei Nutzung der Paging-Dateien werden nur die Volume-Sets, zu denen die Paging-Platten gehören, auf Fremdbelegung überprüft.

Eine gewisse Systemfunktionalität ist einem ausgezeichneten Volume-Set, dem Control-Volume-Set, innerhalb des SM-Pubset zugewiesen: Dieser Volume-Set dient zur Ablage der wichtigsten Metadaten des SM-Pubsets, wie z.B. sämtliche Konfigurationsdaten, der Benutzerkatalog, Verweise auf die Dateikataloge der einzelnen Volume-Sets, Kataloge für spezielle Objekte wie Jobvariablen, No-Space-Dateien und Dateien auf privaten Datenträgern. Dieser Control-Volume-Set muss deshalb immer vorhanden sein und kann nicht aus dem Pubset entfernt werden.
Bei Ausfall des Control-Volume-Sets gilt der gesamte SM-Pubset als ausgefallen.
Für den Control-Volume-Set sollte deshalb durch den Systembetreuer eine besondere Ausfallsicherheit durch Nutzung von Spiegelfunktionen (DRV oder RAID1) gewährleistet sein.
Ein SM-Pubset, der mit DRV betriebene Volume-Sets enthält, kann jedoch nicht als Shared-Pubset verwendet werden.

### Einrichten und Erweitern von Volume-Sets und SM-Pubsets

SM-Pubsets können durch komplettes Neueinrichten oder durch Generieren aus bereits existierenden SF-Pubsets eingerichtet werden.

SM-Pubsets und die zugehörigen Volume-Sets können zusammen in einem SIR-Lauf eingerichtet werden. Beides kann aber auch in verschiedenen SIR-Läufen erfolgen. SIR-Aktionen, die einen importierten Pubset erfordern (z.B. CREATE-PAGING-FILE), können in diesem Fall für einen einzelnen Volume-Set nicht ausgeführt werden. Das Dienstprogramm SIR ist im Handbuch „Dienstprogramme“ [15] beschrieben.

### SM-Pubset und Net-Storage

Zum Verbinden von SM-Pubsets mit Net-Storage siehe Kapitel „Net-Storage-Verwaltung“ auf Seite 429.

### Generieren von SM-Pubsets aus SF-Pubsets

Damit Benutzer SM-Pubsets für einen bereits vorhandenen Dateibestand nutzen können, müssen bestehende SF-Pubsets in SM-Pubsets umgewandelt werden. Dabei wird es oft wichtig sein, mehrere SF-Pubsets zu einem SM-Pubset zusammenzufassen. Hierfür sind verschiedene Vorgehensweisen möglich. Unabhängig vom gewählten Vorgehen sind folgende grundsätzlichen Aspekte zu beachten:

- Die Kennung des Pubsets ist als Catid Bestandteil des Pfadnamens der im Pubset katalogisierten Objekte. Geht ein SF-Pubset in einen SM-Pubset ein, dessen Kennung sich von der des SF-Pubsets unterscheidet, ändert sich die Adressierung der Objekte.
- Befinden sich auf verschiedenen SF-Pubsets, die in den SM-Pubset eingehen, Objekte, deren Pfadnamen sich lediglich durch die Catid unterscheiden, so ergäben sich bei der Bildung des SM-Pubsets Namenskollisionen, d.h. die Pfadnamen wären nach der Bildung des SM-Pubsets nicht mehr eindeutig. Benutzer, für welche sich nur auf einem der in den SM-Pubset eingehenden SF-Pubsets Dateien befinden, können davon nicht betroffen sein, da die Benutzerkennung Bestandteil des Pfadnamens ist.
- Geht ein SF-Pubset in einen SM-Pubset ein, dessen Kennung mehr Zeichen enthält als die des SF-Pubsets, verlängern sich die Pfadnamen der auf dem SF-Pubset liegenden Objekte. Dabei kann sich eine Pfadnamenüberlänge ergeben.

In die Umwandlung von SF-Pubsets in einen SM-Pubset sind prinzipiell die Verarbeitungsebene, Hintergrundebenen und Backup-Archive einzubeziehen. Auf Grund der Komplexität bei den Hintergrundebenen und den Backup-Archiven, soll hier nur die Vorgehensweise bei der Verarbeitungsebene skizziert werden. Eine detaillierte Beschreibung ist im Handbuch „SMS“ [28] zu finden.

Zur **Anpassung der Verarbeitungsebenen** bestehen folgende zwei Möglichkeiten:

- In-Place-Konvertierung mit SMPGEN

  Die In-Place-Konvertierung der Verarbeitungsebene wird durch das Dienstprogramm SMPGEN (siehe Handbuch „Dienstprogramme“ [15]) ermöglicht. Es gestattet, aus einem oder mehreren SF-Pubsets einen SM-Pubset zu bilden, wobei die SF-Pubsets jeweils in einen Volume-Set überführt werden. Die auf den SF-Pubsets liegenden Benutzerdateien bleiben bei der Zusammenführung unverändert erhalten, die Metadaten-Dateien der SF-Pubsets (wie Dateikataloge, Benutzerkataloge, Guards-Kataloge usw.) werden in entsprechende Metadaten-Dateien des SM-Pubsets umgewandelt. Bestimmte Pubset- und Volume-Set-Merkmale, Benutzerkontingente, usw. werden für den SM-Pubset durch SMPGEN automatisch ermittelt. Falls der Systembetreuer davon abweichende Einstellungen wünscht, muss er den SM-Pubset nach der Generierung entsprechend nachbehandeln.

  Zur In-Place-Konvertierung bietet SMPGEN die folgenden Anweisungen an:
  - CREATE-SYSTEM-MANAGED-PUBSET für Prüfung und Konvertierung in einen neu zu erstellenden SM-Pubset
  - MODIFY-SYSTEM-MANAGED-PUBSET für Prüfung und Aufnahme zusätzlicher SF-Pubsets in einen bereits existierenden SM-Pubset

- Sichern und Wiedereinspielen von Dateien

  Es erfolgt eine Vollsicherung der SF-Pubsets, die in den SM-Pubset eingehen (ggf. ohne die auf Hintergrundebenen liegenden Dateien). Anschließend wird mit SIR der SM-Pubset erzeugt. Auf dem SM-Pubset werden für die bisherigen Benutzer der SF-Pubsets Benutzereinträge angelegt. Dies wird durch HSMS unterstützt, indem Sicherungen von Benutzerkatalogen von SF-Pubsets auf SM-Pubsets wieder eingespielt werden können. Dabei werden die für SM-Pubsets eingeführten Benutzerattribute (z.B. Benutzerkontingente für hochverfügbare Dateien) so voreingestellt, dass eine möglichst große Kompatibilität gegeben ist. Anschließend wird der Guards-Katalog eingerichtet, wobei auf SF-Pubsets erstellte Sicherungen benutzt werden können. Schließlich werden die Benutzerdateien wieder eingespielt.

  Der Übergang von SF-Pubsets auf SM-Pubsets durch Sichern und Wiedereinspielen von Dateien ist vor allem dann zu empfehlen, wenn für den SM-Pubset eine andere interne Struktur gewünscht wird, als sie sich durch In-Place-Konvertierung ergeben würde. Dies ist zum Beispiel der Fall, wenn ein SF-Pubset mit vielen Volumes in einen SM-Pubset mit mehreren Volume-Sets übergeführt werden soll.

# 9.4 Pubset-Eigenschaften

## 9.4.1 Formatierungseigenschaften

Die Formatierungseigenschaften eines SF-Pubsets bzw. Volume-Sets werden bereits bei der Initialisierung festgelegt und sind über die gesamte Dauer des Bestehens des SF-Pubsets/Volume-Sets unveränderbar. Darüber hinaus muss bzgl. der Formatierungseigenschaften Homogenität innerhalb des SF-Pubsets/Volume-Sets herrschen, d.h. alle seine Volumes (Platten) haben jeweils die gleiche Formatierungseigenschaft.

Die Formateigenschaft eines SF-Pubsets/Volume-Sets wird von der Systembetreuung bei der Formatierung der Platten (mit SIR bzw. VOLIN) festgelegt:

- Dateiverarbeitung mit und ohne Nutzung des Pamkeys im Bearbeitungsmodus: K und NK
- minimale Allokierungseinheit (min. AU), mit der das DVS Platzzuweisungen für Dateien auf den Platten des Pubsets vornimmt: 6 KB, 8 KB oder 64 KB
- minimale E/A-Transporteinheit (min. TU), in der die Platten-Ein-/Ausgabe erfolgen soll: 2 KB oder 4 KB

Zu den aufgeführten Plattenformaten gibt es die entsprechenden Pubset-Formate, die für importierte Pubsets mit dem Kommando SHOW-PUBSET-CONFIGURATION abgefragt werden können.

Folgende Kombinationen für Plattenformate sind generell möglich:

| Plattenformat | Bearbeitungs-modus | min. AU (in KB) | min. TU (in KB) | max. Plattenanzahl pro SF-Pubset/Volume-Set |
|---|---|---|---|---|
| K-Format | K | 6 | 2 | 32 |
| NK2-Format | NK | 6 | 2 | 32 |
| NK2-Format | NK | 8 | 2 | 255 |
| NK4-Format | NK | 8 | 4 | 255 |
| NK2-Format | NK | 64 | 2 | 255 |
| NK4-Format | NK | 64 | 4 | 255 |

### 9.4.2 Eigenschaftsprofile für die Volume-Set-Selektion

Für SM-Pubsets existieren neben den Formatierungseigenschaften noch weitere Eigenschaften, die vom System bei der Auswahl desjenigen Volume-Sets berücksichtigt werden, das sich als Ablageort für eine Datei jeweils am besten eignet (Volume-Set-Selektion). Im Gegensatz zu den Formatierungseigenschaften werden diese Eigenschaften jedoch durch den Systembetreuer erst dann festgelegt, wenn der Pubset gebildet oder ein freier Volume-Set im Rahmen der Rekonfiguration in den SM-Pubset aufgenommen wird.

**Verfügbarkeit**

Die Verfügbarkeit eines Volume-Sets ist durch das Attribut AVAILABILITY charakterisiert. Mögliche Werte sind STD und HIGH. Zum Erreichen der erhöhten Verfügbarkeit eines Volume-Sets sind verschiedene Mittel gegeben, z.B. Einsatz eines geeigneten RAID-Levels im Plattenspeichersystem (RAID 1/0, RAID6) oder Volume-Spiegelung mit DRV.
Bei der Wertzuweisung für das Volume-Set-Attribut AVAILABILITY mit dem Kommando MODIFY-PUBSET-DEFINITION-FILE wird nicht überprüft, ob die tatsächliche Eigenschaft des Volume-Sets den zugeordneten Werten entspricht. Es obliegt dem Systembetreuer, der Realität entsprechende Werte zu wählen. Dass die Zuordnung des AVAILABILITY-Attributs in den Verantwortungsbereich des Systembetreuers gelegt wird und nicht automatisch durch das System erfolgt, hat folgende Gründe:

- Dem Systembetreuer soll es möglich sein, eigene Erfahrungen in die Bewertung der Verfügbarkeit einfließen zu lassen. Eine durch das System vorgenommene (starre) Bewertung könnte mit der Einschätzung des Systembetreuers kollidieren.
- Es ist sinnvoll, die Bewertung eines Mediums in Relation zu den Eigenschaften der übrigen in einem Pubset vorhandenen Datenträger vorzunehmen.
  Bei der Aufnahme neuer Hardware in einen Pubset kann eine Anpassung der bisherigen Bewertungen notwendig werden.
- Der vom Systembetreuer zugewiesene Verfügbarkeitswert entspricht der langfristigen Positionierung, die dem Volume-Set innerhalb des SM-Pubsets zugedacht ist.
  Eine Änderung des AVAILABILITY-Werts von HIGH zu STD ist mit erheblichem organisatorischem Aufwand verbunden, da zuvor insbesondere alle auf dem Volume-Set liegenden Dateien mit dem Dateiattribut AVAILABILITY=*HIGH auf einen anderen hoch-verfügbaren Volume-Set gebracht werden müssen. Der Verfügbarkeitswert wird sinnvollerweise nur dann modifiziert, wenn sich die Positionierung des Volume-Sets langfristig ändert, nicht jedoch um auf vorübergehende Schwankungen (z.B. bei kurzfristiger Änderung des Betriebszustands eines Volumes von „Dual“ nach „Mono“ bei DRV) zu reagieren. Die explizite Zuordnung bei der Pubset-Generierung bzw. Pubset-Rekonfiguration soll den langfristigen Charakter dieser Festlegung verdeutlichen.

### Performance

Die Beschreibung des Performance-Profils eines Volume-Sets erfolgt durch die Angabe eines Performance-Spektrums und die Spezifikation von Einschränkungen bzgl. der Schreibsicherheit im Kommando MODIFY-PUBSET-DEFINITION-FILE (EDIT-PUBSET-DEFINITION-FILE). Diese komplexe Struktur ist notwendig, um das Verhalten von Volume-Sets mit Caches für die Volume-Set-Selektion ausreichend charakterisieren zu können.

- Performance-Spektrum:
  Ein mit einem Cache ausgestatteter Volume-Set kann i.A. ein Spektrum unterschiedlicher Performance-Anforderungen von seiten der Benutzer abdecken (STD, HIGH, VERY-HIGH). Er ist damit gleichzeitig als Ablageort für Dateien mit der Anforderung PERFORMANCE=*STD, *HIGH oder *VERY-HIGH geeignet. Die unterschiedlichen Performance-Werte korrespondieren mit verschiedenen Arten der Cache-Nutzung, welche bei der Bearbeitung einer Datei möglich sind:

  | | |
  |---|---|
  | PERFORMANCE=*STD | keine Cache-Nutzung |
  | PERFORMANCE=*HIGH | Caching mit Verdrängung |
  | PERFORMANCE=*VERY-HIGH | Caching ohne Verdrängung |

  Die Art der Cache-Nutzung kann individuell den spezifischen Performance-Anforderungen der einzelnen Dateien angepasst werden.

  
  Die gleichzeitige Zuordnung mehrerer unterschiedlicher Caches zu einem Volume-Set ist nicht möglich.

- Schreibsicherheit:
  Die verschiedenen Cache-Medien (Cache-Steuerung, Globalspeicher, Hauptspeicher) unterscheiden sich im Hinblick auf die Schreibsicherheit (WRITE-CONSISTENCY=*BY-CLOSE/*IMMEDIATE). Ein Cache gilt dann als schreibsicher, wenn für Ausgabedaten, die im Cache zwischengepuffert werden, kein höheres Ausfallrisiko besteht, als wenn sie unmittelbar auf Platte geschrieben würden. Nicht-schreibsichere Caches dürfen generell für die Performance-Verbesserung bei lesenden Dateizugriffen verwendet werden. Für Schreibzugriffe sind sie aber nur dann nutzbar, wenn der Benutzer Einschränkungen hinsichtlich Schreibsicherheit akzeptiert und dies durch das Dateiattribut DISK-WRITE=*BY-CLOSE zum Ausdruck bringt.

Performance-Erhöhungen lassen sich neben der Nutzung von Caches auch durch die Verwendung von Volume-Sets erreichen, die aus speziellen, hochperformanten Volumes (im GS emulierte Volumes) gebildet sind. Das Performance-Spektrum solcher Volume-Sets ist auf einen einzigen Performance-Wert reduziert (i.A. VERY-HIGH), der ohne Einschränkung hinsichtlich der Schreibsicherheit gilt.

### Empfehlungen für das Performance-Profil

Analog wie das Verfügbarkeitsprofil wird das Performance-Profil eines Volume-Sets nicht automatisch aus der physikalischen Konstellation ermittelt. Es obliegt dem Systembetreuer mit Hilfe des Kommandos MODIFY-PUBSET-DEFINITION-FILE (EDIT-PUBSET-DEFINITION-FILE), das Performance-Profil der Volume-Sets so festzulegen, dass es die realen Verhältnisse richtig beschreibt. Bei der Beschreibung des Performance-Spektrums muss vor allem darauf geachtet werden, dass die einzelnen Volume-Sets im Vergleich zu den übrigen im Pubset vorhandenen Volume-Sets korrekt eingestuft werden.

Die in folgender Tabelle vorgeschlagenen Einstellungen für die verschiedenen Cache-Medien können folglich hinsichtlich des Performance-Spektrums nur als Orientierungshilfe dienen, da sie die Gegebenheiten konkreter Konfigurationen nicht berücksichtigen. Für die Festlegung der Schreibsicherheit bei erhöhter Performance sind hingegen Empfehlungen möglich, die in jeder Konfiguration anwendbar sind.

| Volume-Set-Konfiguration | empfohlenes Performance-Profil | |
|---|---|---|
| | Performance-Spektrum (PERFORMANCE) | Schreibsicherheit (WRITE-CONSISTENCY) bei erhöhter Performance (PERF=HIGH/VERY-HIGH) |
| Volume-Set, der aus Volumes ohne besondere Performance-Eigenschaften besteht und dem kein Cache zugeordnet ist | STD | ohne Bedeutung |
| Volume-Set, dem ein Hauptspeicher als Cache-Medium zugeordnet ist | STD, HIGH, VERY-HIGH | nicht gegeben (BY-CLOSE) |
| Volume-Set, dem ein flüchtiger Globalspeicher zugeordnet ist (VOLATILITY=*YES) | STD, HIGH, VERY-HIGH | nicht gegeben (BY-CLOSE) |
| Volume-Set, dem ein nicht-flüchtiger Globalspeicher zugeordnet ist (VOLATILITY=*NO) | STD, HIGH, VERY-HIGH | gegeben (IMMEDIATE) |
| Volume-Set, der aus im Globalspeicher emulierten Volumes besteht | VERY-HIGH | gegeben (IMMEDIATE) |
| Volume-Set, dessen Volumes an eine Cache-Steuerung angeschlossen sind | STD, HIGH | gegeben (IMMEDIATE) |

Tabelle 23: Empfehlungen für Performance-Profile

### 9.4.3 Pubset-globale Einstellungen

Pubset-globale Einstellungen dienen vorwiegend zur Steuerung des Pubset-Betriebs.
Mit ihnen können z.B. der Betriebsmodus (shared oder exklusiv) oder funktionseinheiten-spezifische Steuerungsparameter festgelegt werden.

#### CMS-Einstellungen

Sie bestimmen im Wesentlichen Anzahl und Lage der CMS-Puffer, die Einfluss auf den Betrieb der Katalogverwaltung nehmen.

Angaben über die CMS-Puffer können mit Hilfe der Operanden RESIDENT-BUFFERS und NUMBER-OF-BUFFERS der Kommandos ADD-/MODIFY-MASTER-CATALOG-ENTRY oder IMPORT-PUBSET gemacht werden.

Die Puffervereinbarung wird erst beim Importieren des Pubsets (Kommando IMPORT-PUBSET) nach folgender Hierarchie ausgewertet:

1. Explizite Parameterangabe im Kommando IMPORT-PUBSET
2. Angaben über das Kommando ADD- bzw. MODIFY-MASTER-CATALOG-ENTRY. Wird nur einer der Parameter RESIDENT-BUFFERS oder NUMBER-OF-BUFFERS angegeben, gilt für den anderen jeweils der Standardwert. Wird keiner dieser Parameter angegeben, bleiben beide Werte undefiniert.
3. Vereinbarungen laut Systemparameter CATBUFR und BMTNUM
4. Standardwerte (RESIDENT-BUFFERS=*NO, NUMBER-OF-BUFFERS=32)

#### Betriebsmodi

Ein Pubset kann im exklusiven Zugriffsmodus von einem System oder als Shared-Pubset von mehreren Systemen gleichzeitig benutzt werden.
Weiter besteht die Möglichkeit, den Pubset exklusiv für einen Benutzer in Betrieb zu nehmen; d.h. er kann für einen einzelnen Benutzer (eine Benutzerkennung) reserviert werden. Solange diese Reservierung besteht, dürfen nur diese Kennung und die Systembetreuung unter der Kennung TSOS auf diesen Pubset zugreifen.

Die Betriebsmodi können über die Kommandos ADD-/MODIFY-MASTER-CATALOG-ENTRY eingestellt oder – mit Ausnahme der exklusiven Nutzung für einen Benutzer – direkt beim Importieren des Pubsets (Kommando IMPORT-PUBSET) festgelegt werden.

### Maximale Ein-/Ausgabe-Länge

Die max. Ein-/Ausgabe-Länge ergibt sich für einen SM- wie für einen SF-Pubset aus der kleinsten Ein-/Ausgabe-Länge aller zum Pubset gehörenden Volumes, ist also abhängig vom Gerätetyp.

Mit dem Kommando SHOW-PUBSET-CONFIGURATION lässt sich der Wert für die maximale Ein-/Ausgabe-Länge für SF-Pubsets ausgeben. Für SM-Pubsets erfolgt die Anzeige im gleichen Kommando Volume-Set-spezifisch.

### Standard-Einstellungen für Pubset-Space und Dateiformat

Für SM- und SF-Pubsets kann der Standardwert für die Größe von Primär-, Sekundär- und Maximalallokierung der Dateien gesteuert werden (Kommando MODIFY-PUBSET-SPACE-DEFAULTS (EDIT-PUBSET-SPACE-DEFAULTS)).

Für einen SM-Pubset kann mit dem Kommando MODIFY-PUBSET-SPACE-DEFAULTS ...,PUBSET-TYPE=*S-M(FILE-FORMAT=*K/*NK2/*NK4) der Standardwert für das Dateiformat gesteuert werden.

### 9.4.4 Volume-Set-spezifische Einstellungen

Für die Volume-Sets eines SM-Pubsets können spezielle Einstellungen bzgl. Nutzungsart, Nutzungseinschränkungen usw. vorgenommen werden.

**Nutzungsarten für Volume-Sets**

Mit Hilfe des Kommandos MODIFY-PUBSET-DEFINITION-FILE, Operand USAGE, kann die Nutzungsart für die Volume-Sets eines SM-Pubsets festgelegt werden. Es sind folgende alternativen Nutzungsarten möglich:

- STD
  Normalfall. Für den Control-Volume-Set ist STD die einzig mögliche Nutzungsart.
- WORK
  Nur Volume-Sets mit dieser Nutzungsart dienen als Ablageort für Arbeitsdateien.
- HSMS-CONTROLLED
  Volume-Sets mit dieser Nutzungsart werden benötigt, um mit Hilfe von HSMS die S1-Ebene einrichten zu können. Sie dienen ausschließlich diesem Zweck.

**Nutzungseinschränkungen für Volume-Sets**

Die Nutzungseinschränkungen sind im Gegensatz zur Nutzungsart auch kurzfristig änderbar. Es wird in Einschränkungen für das Neuanlegen von Dateien, für das Bearbeiten von Dateien und für das temporäre Stillegen eines Volume-Sets unterschieden.

*Einschränkungen für das Neuanlegen von Dateien*

Das Anlegen neuer Dateien auf einem Volume-Set kann durch das Kommando MODIFY-PUBSET-RESTRICTIONS ..., NEW-FILE-ALLOCATION(MODE=...) eingeschränkt werden.
Das Zurücknehmen einer Einschränkung (mit NEW-FILE-ALLOCATION=*NOT-ALLOWED/*PHYSICAL-ONLY) ist nur zulässig, wenn der Zugang zum Volume-Set nicht eingeschränkt ist. Folgende Einstellungen für MODE sind möglich:

- NOT-RESTRICTED
  Keine Einschränkung.
- PHYSICAL-ONLY
  Dateien können nur durch physikalische Allokierung angelegt werden.
- NOT-ALLOWED
  Es können generell keine neuen Dateien angelegt werden. Diese Einstellung ist primär als Hilfsmittel gedacht, um einen Volume-Set vor dem Entfernen aus dem Pubset leeren zu können. Für den Control-Volume-Set ist sie nicht möglich, da sie zu unerwünschten Behinderungen von Systemfunktionen führen würde.

*Einschränkungen für das Bearbeiten von Dateien*

Das Bearbeiten von Dateien, die auf einem bestimmten Volume-Set liegen, kann durch die Volume-Set-Zugangssperre eingeschränkt werden. Sie ist nur erlaubt, wenn NEW-FILE-ALLOCATION=*NOT-ALLOWED gesetzt ist. Folgende Einstellungen des Kommandos MODIFY-PUBSET-RESTRICTIONS ...,VOLUME-SET-ACCESS(MODE=...) sind möglich:

- ADMINISTRATOR-ONLY
  TSOS-privilegierte Tasks dürfen Dateien uneingeschränkt öffnen. Tasks, welche nicht über das TSOS-Privileg verfügen, sollen die Möglichkeit haben, laufende Bearbeitungen noch geordnet zu beenden, auch wenn sie dazu noch weitere Dateien öffnen müssen. Der Beginn neuer Bearbeitungen soll aber nach Möglichkeit verhindert werden. Tasks ohne TSOS-Privileg dürfen deshalb neue Dateien nur dann öffnen, wenn sie bereits andere Dateien geöffnet haben.
- NOT-RESTRICTED
  Keine Einschränkung.

*Einschränkungen für das temporäre Stilllegen eines Volume-Sets*

Auf einen Volume-Set, der als defekt erkannt worden ist und aus dem SM-Pubset entfernt werden soll, dürfen keine Ein-/Ausgaben erfolgen. Gleiches gilt auch für einen Volume-Set, der nur vorübergehend funktionsunfähig ist, aber mit großer Wahrscheinlichkeit zu einem späteren Zeitpunkt ohne Verlust von Daten wieder benutzt werden kann.
Dies lässt sich erreichen, indem der Volume-Set mit der Einschränkung PROCESSING-STATE(MODE=*HOLD) im Kommando MODIFY-PUBSET-RESTRICTIONS versehen wird. Er geht dadurch in den Konfigurationszustand „in hold“ über.
Für den Control-Volume-Set ist der Zustand „in hold“ nicht möglich.

Der Zustand „in hold“ kann automatisch durch das System oder explizit durch den Systembetreuer herbeigeführt werden. Er wird aufgehoben, indem der Systembetreuer den Volume-Set physikalisch aus dem SM-Pubset entfernt oder die Einschränkung „in hold“ explizit rücksetzt, wenn ein vorübergehend funktionsunfähiger Volume-Set wieder nutzbar ist. Der Systembetreuer sollte darauf achten, dass der Zustand „in hold“ nur kurz andauert, da dadurch die DVS-Funktionalität für den gesamten SM-Pubset beeinträchtigt wird (z.B. Nebeneffekte beim Anlegen neuer Dateien, unvollständige SHOW-FILE-ATTRIBUTES-Ausgabe).

Volume-Sets, die im Zustand „in hold“ exportiert wurden, werden bei der Pubset-Inbetriebnahme implizit wieder aktiviert. Sind weiterhin Volume-Sets funktionsunfähig, wird die Pubset-Inbetriebnahme abgebrochen. Ein Import ist dann nur möglich, wenn der Systembetreuer nicht-reparable Volume-Sets als defekte Volume-Sets entfernen lässt oder sie in den Zustand „in hold“ versetzt (Kommando IMPORT-PUBSET, Operanden DEFECT-VOLUME-SET bzw. IN-HOLD-VOLUME-SET).

### Nutzungseinschränkungen für einzelne Volumes

Für die einzelnen Volumes eines SF-Pubsets oder eines Volume-Sets können Allokierungseinschränkungen erlassen werden (Kommando MODIFY-PUBSET-RESTRICTIONS, Operand ALLOCATION-ON-VOLUME(MODE=...)).
Durch das Setzen der volume-bezogenen Einschränkungen wird die Anzahl der frei verfügbaren Volumes und damit des frei verfügbaren Platzes reduziert. Es wird nicht erlaubt, wenn bereits ein gravierender Speicher-Engpass vorliegt (Überschreitung des Sättigungs-Levels 4) oder durch das Setzen die Restriktion herbeigeführt würde.
Folgende Angaben für MODE sind möglich:

- NOT-RESTRICTED
  Keine Einschränkung.
- PHYSICAL-ONLY
  Auf dem Volume wird nur Platz vergeben, wenn er explizit durch physikalische Allokierung angefordert wird; es erfolgen keine Sekundärallokierungen.
- NOT-ALLOWED
  Es erfolgen generell keine Allokierungen.

### Sättigungsschwellwerte

Den einzelnen SF-Pubsets bzw. Volume-Sets eines SM-Pubsets sind durch den Systembetreuer festlegbare Schwellwerte für die Überwachung von Speicherengpässen zugeordnet (Kommando SET-/MODIFY-SPACE-SATURATION-LEVEL; siehe auch SPACEPRO (Seite 378)).

Die eigentliche Sättigungsüberwachung erfolgt für Volume-Sets analog wie für SF-Pubsets. Der höchste überschrittene Sättigungs-Level bestimmt den aktuellen Sättigungszustand, siehe dazu auch den Abschnitt „Überwachung der Speicherplatz-Sättigung“ auf Seite 393.

### 9.4.5 Storage-Klassen, Management-Klassen und Volume-Set-Listen

Aus Benutzersicht erscheint ein SM-Pubset als ein Dateibehälter, der bestimmte Services für die Dateiverwaltung bereitstellt. Diese werden als Datei-Services bezeichnet. Sie lassen sich in Storage-Services (z.B. gebotene Performance, Verfügbarkeit) und HSMS-Management-Services (z.B. Steuerung der Dateiverdrängung auf Hintergrundebenen, Steuerung der Erstellung von Dateisicherungen) unterteilen. Das Anfordern der gewünschten Datei-Services durch die Benutzer erfolgt mit Hilfe von Dateiattributen, für welche den Dateien passende Werte zugewiesen werden müssen.

#### Storage-Services

Die Anforderung der Storage-Services kann durch folgende alternative Möglichkeiten erfolgen:

a) Direktattributierung

Bei der Direktattributierung spezifiziert der Benutzer für eine Datei die Werte für die ablageort-relevanten Attribute Performance, Verfügbarkeit, usw. jeweils einzeln (siehe Kommandos CREATE-FILE oder MODIFY-FILE-ATTRIBUTES).

b) Zuordnung einer Storage-Klasse

Der Systembetreuer kann Storage-Klassen einrichten, welche für die ablageort-relevanten Dateiattribute eine bestimmte Kombination von Attributwerten repräsentieren (siehe Kommandos CREATE-STORAGE-CLASS und MODIFY-STORAGE-CLASS (EDIT-STORAGE-CLASS)).

Der Benutzer kann für eine Datei aus den vorhandenen Storage-Klassen diejenige aussuchen, welche den jeweiligen Anforderungen am besten entspricht, und sie der Datei zuordnen (siehe Kommandos CREATE-FILE oder MODIFY-FILE-ATTRIBUTES, Operand STORAGE-CLASS).

Der vom Benutzer für eine Datei gewünschte Storage-Service wird berücksichtigt, wenn für die Datei der Ablageort in der Verarbeitungsebene bestimmt wird. Dies ist der Fall, wenn eine Datei neu angelegt wird, wenn eine Datei von einer Hintergrundebene in die Verarbeitungsebene zurückgeholt wird, oder wenn der Benutzer für eine bestehende Datei die ablageort-relevanten Dateiattribute so ändert, dass sie mit dem bisherigen Ablageort unverträglich sind. Dabei ermittelt das System automatisch den am besten geeigneten Volume-Set (Volume-Set-Selektion, siehe Seite 331).

Das Systemverhalten kann dabei durch die Systembetreuung beeinflusst werden, um bestimmte Strategien im Data Center zu realisieren. Das Instrument hierfür sind Volume-Set-Listen. Sie können durch den Systembetreuer eingerichtet und mit Storage-Klassen verknüpft werden (siehe Kommandos CREATE-VOLUME-SET-LIST und CREATE-STORAGE-CLASS, Operand VOLUME-SET-LIST).

Ordnet ein Benutzer einer Datei eine Storage-Klasse zu, der eine Volume-Set-Liste zugewiesen ist, wird die Datei durch das System bevorzugt auf einem Volume-Set aus dieser Liste abgelegt. Belegt die Datei bereits Speicherplatz auf einem anderen Volume-Set, wird sie - wenn möglich - von diesem auf einen der bevorzugten Volume-Sets verlagert.
Dateien, denen keine Storage-Klasse oder eine Storage-Klasse ohne Volume-Set-Liste zugewiesen ist, werden bevorzugt auf solchen Volume-Sets abgelegt, die keiner der vom Systembetreuer eingerichteten Volume-Set-Listen angehören.

Der Systembetreuer kann für jeden Benutzer eine Standard-Storage-Klasse festlegen. Diese kann Direktattribute unwirksam machen, wenn kein Recht auf physikalische Attributierung vorliegt.

### HSMS-Management-Services

Das Anfordern der Management-Services erfolgt mit Hilfe von HSMS-Management-Klassen. Sie müssen vom Systembetreuer eingerichtet werden (HSMS-Anweisung CREATE-MANAGEMENT-CLASS). Die HSMS-Management-Klassen repräsentieren bestimmte Verfahren für Dateisicherung (z.B. Sicherungshäufigkeit, Lebensdauer der Sicherungsversionen, usw.) sowie Regeln, welche die Verdrängung auf die Hintergrundebenen steuern (z.B. Verdrängbarkeit in Abhängigkeit der seit dem letzten Zugriff vergangenen Zeit).

Indem der Benutzer einer Datei eine HSMS-Management-Klasse zuweist (Kommandos CREATE-FILE oder MODIFY-FILE-ATTRIBUTES, Operand MANAGEMENT-CLASS), unterwirft er sie den Sicherungs- bzw. Verdrängungsverfahren, welche durch diese HSMS-Management-Klasse repräsentiert werden.
Für den speziellen Benutzerwunsch, bestimmte Dateien generell oder unter bestimmten Randbedingungen von der Verdrängung auf Hintergrundebenen auszuschließen, stehen zusätzlich Migrationssperren zur Verfügung (Kommandos CREATE-FILE bzw. MODIFY-FILE-ATTRIBUTES, Operand MIGRATE). Ggf. muss der Benutzer dazu die Erlaubnis zur physikalischen Allokierung besitzen.

Die Realisierung des Services, den eine HSMS-Management-Klasse repräsentiert, unterscheidet sich von der Realisierung der Storage-Services dadurch, dass sie nicht durch Automatismen erfolgt, die bereits in das System integriert sind. Dem Systembetreuer werden an Stelle dessen automatisierungsfreundliche Schnittstellen zur Verfügung gestellt, mit denen er Kommandoprozeduren oder Programme entwickeln kann, welche die vorgesehenen Verfahren abwickeln.

## 9.4.6 Physikalische Merkmale von Volume-Sets

Wie bereits erwähnt, muss der Systembetreuer dafür sorgen, dass die Eigenschaftsprofile der Volume-Sets durch die physikalische Volume-Set-Konfiguration erfüllt werden. Hierfür stehen auf physikalischer Ebene eine Vielzahl von Möglichkeiten zur Verfügung. Es folgt die Darstellung einzelner, exemplarischer Aspekte:

**Physikalische Eigenschaften der einzelnen Volumes**

Die physikalischen Eigenschaften eines Volume-Sets sind durch die Eigenschaften der ihm zugehörigen Volumes sowie deren I/O-Konfiguration (Controller, Zugangspfade, usw.) bestimmt. Damit den Volume-Sets sinnvolle Eigenschaftsprofile zugeordnet werden können, sollte die Volume-Konfiguration der Volume-Sets homogen sein, d.h. der Systembetreuer sollte nur Volumes zu einem Volume-Set zusammenfassen, deren Eigenschaften mit dem vorgesehenen Eigenschaftsprofil harmonieren. Bei bestimmten Eigenschaften wird die Homogenität des Volume-Sets durch das System erzwungen (z.B. Größe der Allokierungseinheit, Format, GS-emulierte Volumes).

**Maximale Ein-/Ausgabe-Länge**

Eine der physikalischen Eigenschaften, die ein Volume charakterisieren, ist seine max. Ein-/Ausgabe-Länge, d.h. die max. Anzahl von Blöcken, die bei einem I/O-Auftrag übertragen werden können. Sie wird durch den Gerätetyp bestimmt.
Der kleinste Wert aller einem Volume-Set angehörenden Volumes bestimmt die max. Ein-/Ausgabe-Länge des Volume-Sets.
Wird in einen Volume-Set mit Hilfe der Pubset-Rekonfiguration ein weiteres Volume hinzugenommen, kann sich also u.U. die max. Ein-/Ausgabe-Länge des Volume-Sets verkleinern. Die max. Ein-/Ausgabe-Länge eines Volume-Sets wird durch bestimmte privilegierte Anwendungen (ARCHIVE, CCOPY) zur Optimierung der Ein-/Ausgaben herangezogen.

Mit dem Kommando SHOW-PUBSET-CONFIGURATION kann die maximale Ein-/Ausgabe-Länge eines Volume-Sets ausgegeben werden.

### Volume-Sets als unabhängige Ausfalleinheiten

Die Pubset-Verwaltung unterstützt die Möglichkeit, dass die einzelnen Volume-Sets eines SM-Pubsets voneinander unabhängige Ausfalleinheiten darstellen, d.h. bei dem Ausfall eines Volumes ist nur der Volume-Set betroffen, dem dieses angehört. Falls es sich dabei nicht um den Control-Volume-Set handelt, kann der restliche SM-Pubset weiterhin in Betrieb bleiben.

Allerdings sind dafür auch Maßnahmen auf physikalischer Ebene erforderlich:
Volume-Sets, die bezüglich des Ausfalls unabhängig sein sollen, müssen auf physikalische Konfigurationen abgebildet werden, die hinsichtlich Ausfall unabhängig voneinander sind. Beispielsweise müssen die Zugangspfade (Kanäle, Controller, usw.) zu den Volumes von Volume-Sets, die bezüglich Ausfallsicherheit unabhängig sein sollen, voneinander entkoppelt sein. Auch die den Volume-Sets zugeordneten Cache-Konfigurationen sind zu berücksichtigen.

### DRV für Volume-Sets

Zum Bereitstellen besonders ausfallsicherer Volume-Sets können DRV-Funktionen genutzt werden. Diese bieten für SM-Pubsets auf Volume-Set-Ebene eine analoge Bedienoberfläche wie für SF-Pubsets auf Pubset-Ebene. Shared-Pubsets können nicht mit DRV betrieben werden.

### 9.4.7 Pubset-Caches

Im Hiperfile-Konzept (siehe Abschnitt „PFA: Performant File Access“ auf Seite 291) kann zur Verbesserung der Datenzugriffszeiten pro Volume-Set oder SF-Pubset ein Cache vorgeschaltet werden, in dem die I/O-Daten zwischengepuffert werden. Der Datentransfer erfolgt (bei Cache-Hit) in wesentlich höherer Geschwindigkeit. Bei Schreibzugriffen gilt dies immer, der Transfer auf die jeweiligen Platten wird dann asynchron abgearbeitet. Bei den zwischengeschalteten Caches handelt es sich generell um Schreib-/Lese-Caches, deren wesentliches Unterscheidungsmerkmal ist, ob es sich um einen flüchtigen oder nicht-flüchtigen Cache handelt. Als Cache-Medien stehen zur Verfügung:

- Hauptspeicher
  Der Hauptspeicher ist als flüchtiges Medium für Lese-Caches oder für flüchtige Schreib-Caches nutzbar.
- Globalspeicher
  Je nach Ausbauart (mit oder ohne eigener Stromversorgung) kann der Globalspeicher ein nicht-flüchtiges oder flüchtiges Cache-Medium sein.
- Plattenspeichersysteme
  Plattenspeichersysteme verfügen standardmäßig über eigene Cache-Bereiche. Diese werden aber **nicht** im HIPERFILE/PFA-Konzept verwaltet.

#### Bestimmung der Cache-Konfiguration

Voraussetzung für die Nutzung von Caches ist das Bereitstellen bestimmter Cache-Konfigurationen. Für SM-Pubsets werden hierzu auf Volume-Set-Ebene analoge Möglichkeiten geboten, wie bei SF-Pubsets auf Pubset-Ebene.

Die erforderlichen Maßnahmen für die Nutzung von Caches sind stark vom jeweiligen Cache-Medium abhängig. Im Folgenden werden die Merkmale der Cache-Konfiguration eines Volume-Sets vor allem im Hinblick auf Globalspeicher-Caches (GS-Caches) skizziert.

Für jedes Volume-Set ist durch den Systembetreuer festzulegen, ob es mit einem Cache betrieben werden soll, welches Cache-Medium gewünscht wird, und wie groß der Cache-Bereich für das Volume-Set sein soll (Kommando MODIFY-PUBSET-CACHE-ATTRIBUTES ...,CACHE-MEDIUM=...,CACHE-SIZE=...).
Bei Cache-Nutzung sind ferner bestimmte Cache-Betriebsparameter anzugeben, z.B. MODIFY-PUBSET-CACHE-ATTRIBUTES ...,CACHE-MEDIUM=*GLOBAL-STORAGE (CACHE-SEGMENT-SIZE =*4KB,FORCE-OUT=*AT-HIGH-FILLING). Der Umfang und die Bedeutung der Betriebsparameter sind für die einzelnen Cache-Medien unterschiedlich. Die Angaben des Systembetreuers bestimmen die defined-Werte für die Cache-Konfiguration der Volume-Sets. Sie werden für die einzelnen Volume-Sets in der Pubset-Konfigurationsdatei des SM-Pubsets (siehe Seite 350) vermerkt.
Änderungen der defined-Werte sind (im Unterschied zu SF-Pubsets) nur möglich, wenn der Pubset in Betrieb ist.

Die Werte für die Volume-Set-Cache-Konfiguration kommen zum Tragen:

- wenn der Pubset in Betrieb genommen wird (IMPORT-PUBSET)
- wenn ein Volume-Set bisher im Zustand „defined only“ war und ihm durch dynamische Pubset-Rekonfiguration eine physikalische Volume-Set-Konfiguration zugewiesen wird (mit dem Kommando MODIFY-PUBSET-PROCESSING ...,VOLUME-SET-SUPPORT= *ADD(...))
- wenn durch das Kommando START-PUBSET-CACHING der Cache aktiviert wird

Dabei versucht das System, für den Volume-Set entsprechend den eingestellten defined-Werten einen Cache-Bereich einzurichten. In bestimmten Situationen können sich Abweichungen in der Größe ergeben. Die Größe des aktuell zugeordneten Cache-Bereichs sowie die aktuell wirksamen Cache-Betriebsparameter werden durch die current-Werte der (Volume-Set-) Cache-Konfiguration beschrieben.

Das Auflösen des Cache-Bereichs eines Volume-Sets erfolgt:

- wenn der Pubset außer Betrieb genommen wird (EXPORT-PUBSET)
- wenn einem Volume-Set durch dynamische Pubset-Rekonfiguration die physikalische Volume-Konfiguration entzogen wird (Kommando MODIFY-PUBSET-PROCESSING ...,VOLUME-SET-SUPPORT=*REMOVE(...))
- wenn der Cache durch das Kommando STOP-PUBSET-CACHING deaktiviert wird

Außer Betrieb befindliche Pubsets sind im Normalfall nicht mit Cache-Bereichen konnektiert. In Sonderfällen, z.B. nach einem System-Crash können auch exportierten SF-Pubsets bzw. den Volume-Sets von exportierten SM-Pubsets Cache-Bereiche zugeordnet sein.

Es gibt die Möglichkeit, auch für shared importierte Pubsets Cache-Bereiche im Haupt- oder Globalspeicher anzulegen. Dabei handelt es sich nicht um einen globalen Cache-Bereich für alle Pubset-Sharer (wie beim Medium Controller), sondern um system-lokale Cache-Bereiche.
Beim Medium GS werden sowohl globale GS-Konfigurationen in einem HIPLEX (alle Cache-Bereiche der Pubset-Sharer liegen in einer shared genutzten GS-Partition in einem global genutzten GS) als auch lokale GS-Konfigurationen (die Cache-Bereiche liegen in einem/mehreren lokal genutzten Globalspeichern) unterstützt. Die Konfiguration durch MODIFY-PUBSET-CACHE-ATTRIBUTES (EDIT-PUBSET-CACHE-ATTRIBUTES) erfolgt bei SF-Pubsets system-lokal, bei Volume-Sets von SM-Pubsets gelten die Einstellungen für alle Pubset-Sharer. Eine genaue Beschreibung der Einsatzmöglichkeiten ist im Handbuch „DAB“ [10] enthalten.

### GS-Caches

Als Voraussetzung dafür, dass für Volume-Sets eines SM-Pubsets GS-Cache-Bereiche bereitgestellt werden können, müssen durch den Systembetreuer GS-Partitions für den Pubset eingerichtet worden sein. Diese sind SM-Pubset-spezifisch anzulegen. Sie dienen als Reservoir für alle GS-Cache-Bereiche, die für einen der Volume-Sets des SM-Pubsets angelegt werden. Die Korrelation zwischen GS-Partitions und SM-Pubset erfolgt durch eine Namenskonvention für DAB-GS-Partitions (siehe Handbuch „DAB“ [10]).

Die Möglichkeit von pubset-spezifischen GS-Partitions ist auch für SF-Pubsets vorgesehen, d.h. wenn für einen SF-Pubset ein GS-Bereich bereitgestellt werden soll und GS-Partitions mit passendem Namen vorhanden sind, werden diese benutzt. Existiert aber keine passende GS-Partition für den SF-Pubset, wird der GS-Cache-Bereich wie bisher in einer auf den Home-Pubset bezogenen Partition (Home-Partition) desjenigen Systems angelegt, an dem der Pubset (exklusiv) importiert wird. Diese kann gleichzeitig Cache-Bereiche für mehrere SF-Pubsets enthalten.

Caches sind mit einem bestimmten Ausfallrisiko behaftet. Bei einer nur lesenden Dateiverarbeitung können Caches unabhängig von ihrer Zuverlässigkeit als Zwischenpuffer verwendet werden, ohne dass die Ausfallsicherheit der Datei beeinflusst wird. Bei der Zwischenpufferung von schreibenden Zugriffen hingegen führt der Ausfall des Caches zu Datenverlusten, welche die Zerstörung der gesamten Datei zur Folge haben können.
In die Beurteilung der Ausfallsicherheit eines Volume-Sets/SF-Pubsets, der mit einem Cache betrieben wird, muss daher neben der Ausfallsicherheit der Volumes und ihrer Zugangspfade auch die Verfügbarkeit eines zugeordneten Caches einbezogen werden.
Im Fall von GS-Caches sind folgende Aspekte zu beachten:

- Ausfallsicherheit der Stromversorgung für GS-Caches (VOLATILITY)

  GS-Caches können durch spezielle Vorkehrungen (Batterie-Backup, unterbrechungsfreie Stromversorgung usw.) gegen Stromausfall abgesichert werden. Ohne diese Maßnahmen sind die GS-Caches flüchtig, d.h. bei Stromausfall gehen die in ihnen abgelegten Daten verloren. Dateien, für welche Ausgabedaten in den betroffenen GS-Bereichen zwischengepuffert waren, sind anschließend nicht mehr in ihrem aktuellen Bearbeitungszustand erhalten.
  Die Ausfallsicherheit der Stromversorgung wird dem DVS mit dem Kommando MODIFY-PUBSET-CACHE-ATTRIBUTES, Operand VOLATILITY=*NO bekanntgegeben.

- Ausfallsicherheit der Cache-Medien selbst

  Neben der Stromversorgung für GS-Caches sind auch die Cache-Medien selbst mit einem bestimmten Ausfallrisiko behaftet. Durch die Verwendung von redundanter Datenaufzeichnung kann das Ausfallrisiko für Daten, die in einem GS-Cache gepuffert werden, reduziert werden. Redundante Datenaufzeichnung kann für GS-Caches mit dem Kommando MODIFY-PUBSET-CACHE-ATTRIBUTES, Operand GLOBAL-STORAGE-UNIT=*DUAL gefordert werden.

- Ausfallsicherheit bei System-Crash

  GS-Caches bieten generell Ausfallsicherheit bei einem System-Crash, da alle für die Nutzung der Cache-Bereiche erforderlichen Metadaten in dem Cache selbst abgelegt sind.

In einem flüchtigen, d.h. nicht gegen Stromausfall geschützten GS-Cache werden keine Ausgabedaten von Dateien zwischengepuffert, für welche der Benutzer Schreibsicherheit fordert (Dateiattribut DISK-WRITE=*IMMEDIATE).

Im Unterschied zum Cache-Medium Hauptspeicher kann das System bei GS-Caches nicht selbst entscheiden, ob Nicht-Flüchtigkeit gegeben ist.
Es benötigt daher Informationen durch den Systembetreuer, welche dieser über die Festlegung der VOLATILITY (Flüchtigkeit) bereitstellen muss (siehe Kommando MODIFY-PUBSET-CACHE-ATTRIBUTES):
Die Einstellung VOLATILITY=*YES bewirkt, dass der GS-Cache nur zur Lese-Pufferung von Daten oder zur Pufferung von Dateien verwendet wird, für welche der Benutzer eine eingeschränkte Schreibsicherheit akzeptiert (Dateiattribut DISK-WRITE=*BY-CLOSE). Sie ist für einen GS-Cache unbedingt dann zu empfehlen, wenn er über keine ausfallsichere Stromversorgung verfügt.
Dies gilt analog für das Performance-Profil, das für Volume-Sets von SM-Pubsets festgelegt werden kann.

# 9.5 Pubset-Administration

Die Pubset-Administration gliedert man in einen statischen und einen dynamischen Teil. Der statische Teil befasst sich mit der Beschreibung und Administration von Verwaltungsdaten und ihren Datenstrukturen, der dynamische mit der In- und Außerbetriebnahme sowie der Pubset-Rekonfiguration.
Am Ende des Abschnitts wird auf die Auskunftsfunktionen bzgl. der Pubset-Verwaltungsdaten eingegangen.

## 9.5.1 Metadaten und deren Administration

Unter dem Begriff Pubset-Metadaten werden alle Daten verstanden, die letztendlich für den Pubset-Betrieb benötigt werden, also z.B. die physikalischen Konfigurationsdaten, die physikalischen und logischen Eigenschaften sowie die Betriebsparameter (Benutzerkatalog, Benutzerkontingente, usw.). Den prinzipiellen Aufbau aller wichtigen Pubset-Metadaten zeigen Bild 17 (für einen SM-Pubset) und Bild 18 (für einen SF-Pubset).

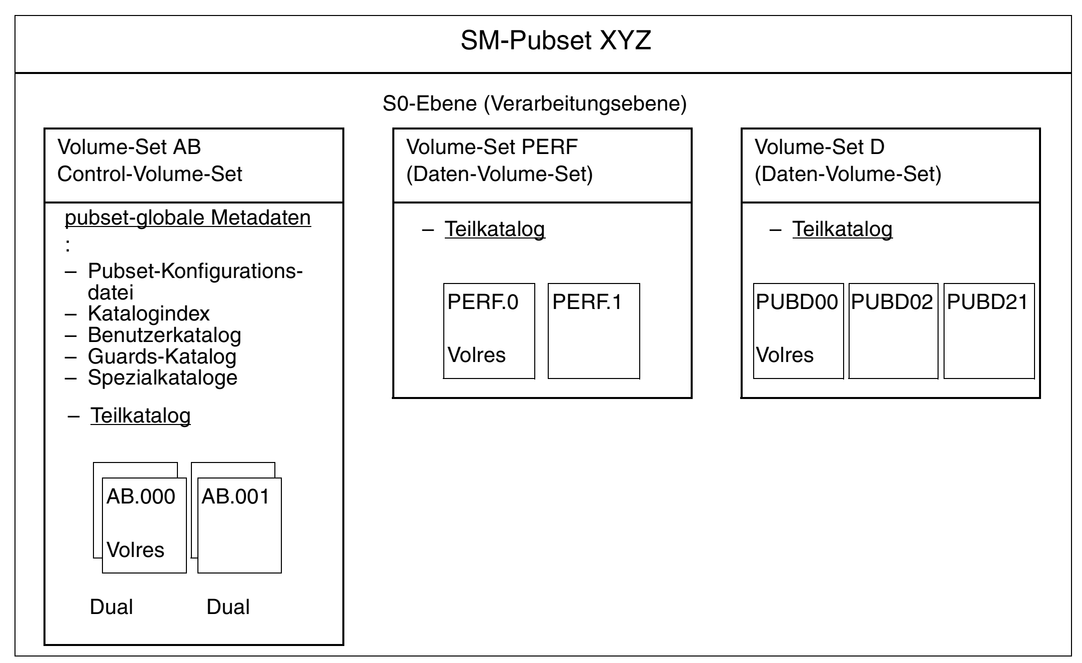


Bild 17: SM-Pubset-Metadaten

Jeder Volume-Set besitzt eine so genannte **Volres**. Die Volres ist die Systemplatte eines Volume-Sets, die dessen Verwaltungsdaten enthält. Die Volres entspricht der Pubres bei SF-Pubsets.

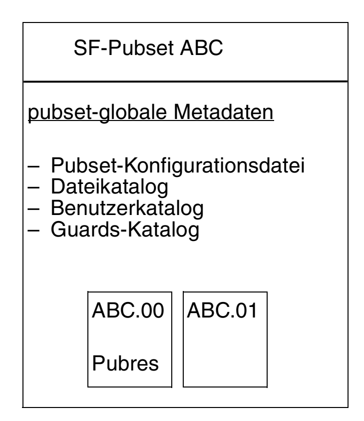



Bild 18: SF-Pubset-Metadaten

Die **Pubres** ist die obligatorische Systemplatte eines SF-Pubset. Sie enthält im SVL ein Verzeichnis der zum Pubset gehörenden Platten und einen Verweis auf den Beginn des TSOSCAT.

#### Konfigurationsdaten und Betriebsparameter

Zur Verwaltung der Konfigurationsdaten und Betriebsparameter sind primär Ablageorte für die einzelnen Verwaltungsdaten erforderlich. Man unterscheidet hierbei zwischen System- und Pubset-spezifischen Ablageorten.

Pubset-spezifische Ablageorte dienen zur Ablage von Konfigurationsdaten und Betriebsparametern, die fest an den Pubset gebunden sind, unabhängig davon, an welchem System der Pubset in Betrieb genommen wird. Dazu gehören z.B. die Volume- bzw. Volume-Set-Konfiguration sowie die physikalischen und logischen Pubset- oder Volume-Set-Eigenschaften. Als Ablageort hierfür dienen das Standard-Volume-Label (SVL) der jeweiligen Volres, einer ausgezeichneten Platte des Volume-Sets, und die Pubset-Konfigurationsdatei eines SM-Pubset.

System-spezifische Ablageorte dienen vorwiegend zur Ablage von Pubset-Betriebsparametern, die speziell für den Betrieb am aktuellen System benötigt werden. Darunter fallen z.B. spezielle Tuning-Werte (siehe Seite 359) für die Pubset-Session. Als system-spezifischer Ablageort von Pubset-Verwaltungsdaten dient das Verzeichnis MRSCAT.

*SVL der Pubres eines SF-Pubsets bzw. der Volres eines Volume-Sets*

Im SVL (Standard-Volume-Label) der Pubres eines SF-Pubsets bzw. der Volres eines Volume-Sets sind alle pubset- bzw. volume-set-globalen Informationen hinterlegt. Die Informationen auf den anderen Platten des SF-Pubsets/Volume-Sets beschränken sich ausschließlich auf Platteneigenschaften und sind deshalb hier nicht von entscheidender Bedeutung.

Der Aufbau des SVL erfolgt in unterschiedlichen Satztypen. Für das DVS sind zwei spezielle Satztypen relevant: Der DMS-Record und der Pubset-Record.

- DMS-Record
  Es sind ein Verweis auf den Dateikatalog TSOSCAT bzw. TSOSCAT.<volume-set-id>, der Wert der Allokierungseinheit, ein Indikator über die normale oder abnormale Beendigung der letzten Pubset-Session, ein Kennzeichen über den Betriebsmodus (lokal oder shared) und Informationen über einen eventuell genutzten Cache hinterlegt.
- Pubset-Record
  Es ist die Plattenkonfiguration (Volume-Katalog) des SF-Pubsets/Volume-Sets hinterlegt. Abhängig von der max. Plattenanzahl ist der Volume-Katalog an unterschiedlichen Stellen hinterlegt:
  - Er ist im Pubset-Record selbst hinterlegt, falls die max. Plattenanzahl 32 beträgt. Der Volume-Katalog im Pubset-Record enthält die VSN, den Volume-Typ der jeweiligen Platte und den Zeitstempel der letzten Betriebsphase des Volumes.
  - Er befindet sich in einem 4KB großen Extrablock auf der Pubres/Volres, wenn die max. Plattenanzahl 255 Platten beträgt. In diesem Fall steht im Pubset-Record lediglich ein Verweis auf diesen Block.

Das Einrichten des gesamten SVL geschieht zum Zeitpunkt der Pubset-Generierung über die Produkte SIR und VOLIN. Dabei werden zunächst alle Daten, soweit sie zu diesem Zeitpunkt noch nicht bekannt sind, mit Standardwerten vorbelegt. Direkte Änderungen der Inhalte sind mit dem Kommando SET-PUBSET-ATTRIBUTES möglich; indirekte Änderungen nur während der physikalischen Rekonfiguration.

Besitzt der Pubset die Eigenschaft „große Volumes“ (Kapazität von ≥ 32 GB), erhält das Versionsfeld des Basis-Records der Pubres bzw. der Volres einen zusätzlichen Wert, der anzeigt, dass der Pubset in BS2000/OSD < V5.0 nicht bedient werden kann.

*Pubset-Konfigurationsdatei*

Die Pubset-Konfigurationsdatei ($TSOS.SYS.PUBSET.CONFIG, Pubset-Configuration-File) eines Pubsets hat zwei Funktionen: Zum einen dient sie zur Ablage sämtlicher Konfigurationsdaten eines SM-Pubsets. Das bedeutet, dass in dieser Datei alle zu einem SM-Pubset gehörenden Volume-Sets mit all ihren Eigenschaften hinterlegt sind. Zum anderen dient die Pubset-Konfigurationsdatei im Fall eines Rekonfigurationsauftrags für den Pubset (siehe Seite 367) als Ablagemedium für die Rekonfigurationsdaten, um bei abnormaler Beendigung des Auftrags diesen wiederholen zu können.

Die Pubset-Konfigurationsdatei ist eine ISAM-Datei und besteht aus drei Satztypen:

- Pubset-Record
  Hier finden sich alle pubset-globalen Daten des SM-Pubset, wie z.B. die Anzahl der Volume-Sets und der Name des Control-Volume-Sets.
- Volume-Set-Record
  Hiervon gibt es der Anzahl der Volume-Sets entsprechend viele Sätze. In diesen sind alle volume-set-spezifischen Betriebsdaten hinterlegt.
- Reconfiguration-Record
  Enthält genau die Informationen, die für eine Fortführung eines unterbrochenen Rekonfigurationsauftrags benötigt werden.

Die Pubset-Konfigurationsdatei wird zum Zeitpunkt der Generierung des SM-Pubsets eingerichtet. Dabei ist egal, ob es sich um eine Neugenerierung oder um ein Zusammenführen mehrerer SF-Pubsets zu einem SM-Pubset handelt. Bei SF-Pubsets wird die Pubset-Konfigurationsdatei nur im Fall des ersten Rekonfigurationsauftrags für den Pubset eingerichtet.

Änderungen der Pubset-Konfigurationsdatei sind nur möglich, wenn der SM-Pubset in Betrieb ist. Explizite Änderungen sind direkt mit dem Kommando MODIFY-PUBSET-DEFINITION-FILE (EDIT-PUBSET-DEFINITION-FILE) möglich. Implizite Änderungen bzgl. des Volume-Set-Status, der Nutzungseinschränkungen usw. werden bei der In- und Außerbetriebnahme oder mit den Kommandos MODIFY-PUBSET-PROCESSING, MODIFY-PUBSET-RESTRICTIONS, MODIFY-PUBSET-CACHE-ATTRIBUTES realisiert.

*Verzeichnis MRSCAT*

Der MRSCAT ist das Verzeichnis aller Pubsets. Er hat zwei wesentliche Funktionen: Zum einen dient er dazu, Pubset-Betriebsparameter permanent, d.h. über die BS2000-Session hinaus, ablegen zu können. Zum anderen dient er während der BS2000-Session als Cache (zur Ablage aller Pubset-Betriebsparameter) im Hauptspeicher und ermöglicht deshalb einen schnellen und einfachen Zugriff auf diese Parameter.

Der MRSCAT dient insbesondere auch allen Funktionseinheiten des DVS zur Ablage ihrer pubset-spezifischen Verwaltungsdaten. Gleichzeitig stellt er das Verzeichnis für alle Dateikataloge dar und gibt Auskunft über deren Verfügbarkeit. Der MRSCAT ist eine zentrale Datenstruktur des DVS und stets Ausgangspunkt jeglicher Dateiadressierung.
Folgendes Bild soll diese Eigenschaft verdeutlichen.

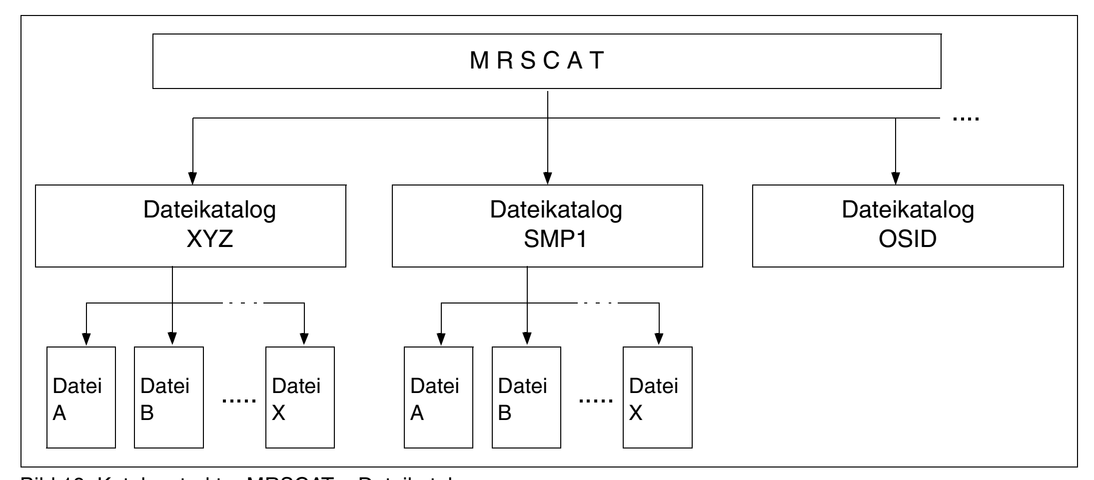



Bild 19: Katalogstruktur MRSCAT – Dateikatalog

*Statische Datenhaltung*

Zur permanenten Ablage der Betriebsparameter dienen die beiden ISAM-Dateien $TSOS.SYSTEM.MRSCAT und $TSOS.SYSTEM.MRSCAT.COPY, die auf dem Home-Pubset abgelegt sind und auf konsistentem Stand gehalten werden, um bei einem eventuellen Ausfall einer der beiden Dateien nach wie vor auf gültige Daten zugreifen zu können. Eingerichtet werden die Dateien bei der Systemeinleitung, wenn ein Pubset erstmalig als Home-Pubset genutzt wird, oder wenn ein Zugriff auf die bestehenden Dateien nicht möglich ist. In diesem Fall werden, um die Systemeinleitung nicht abbrechen zu müssen, die defekten Dateien automatisch gelöscht und zwei neue unter gleichen Namen erstellt.

Ein weiterer Ablageort zur permanenten Ablage von MRSCAT-Informationen ist die erste Seite der Katalogdatei TSOSCAT bzw. TSOSCAT.<control-volume-set-id>. Dort werden für den Home-Pubset und für Control-Volume-Sets von SM-Pubsets die MRSCAT-Informationen des eigenen SF-Pubsets/Volume-Sets hinterlegt. Dieser zusätzliche Ablageort ist erforderlich, da ISAM-Dateizugriffe erst möglich sind, wenn der jeweilige Pubset in Betrieb

ist. Voraussetzung für die Inbetriebnahme des jeweiligen Objekts ist jedoch seine MRSCAT-Information. Zur Vermeidung dieser Deadlock-Situation wird die eigene MRSCAT-Information von Home-Pubsets oder Control-Volume-Sets in der Katalogdatei hinterlegt, auf die bereits sehr frühzeitig mittels CMS-Funktionen zugegriffen werden kann.

*Dynamische Datenhaltung*

Die zweite Funktion des Verzeichnisses MRSCAT ist die performante Ablage aller Pubset-Betriebsparameter während der BS2000-Session im Arbeitsspeicher. Dabei existiert pro Volume-Set, SF-Pubset oder SM-Pubset ein Eintrag im MRSCAT. Als Identifizierer eines MRSCAT-Eintrags dient die Katalogkennung (catid), die der jeweiligen Pubset- bzw. Volume-Set-Kennung entspricht. Die max. Anzahl von MRSCAT-Einträgen ist begrenzt und kann zur Optimierung der Zugriffs-Performance den Wert 128, 256, 512, 1024, 2048 oder 4096 annehmen. Mit dem Systemparameter DMCMAXP kann diese max. Anzahl festgelegt werden.

Möglichkeiten für den Systembetreuer, MRSCAT-Einträge einzurichten, zu modifizieren und zu löschen, werden durch folgende Kommandoschnittstellen angeboten:

- ADD-MASTER-CATALOG-ENTRY
  Neueinrichten von MRSCAT-Einträgen mit gleichzeitigem Initialisieren der statischen Pubset-Parameter mit Ausnahme der Cache-Konfiguration.
- MODIFY-MASTER-CATALOG-ENTRY (EDIT-MASTER-CATALOG-ENTRY)
  Änderung der statischen Pubset-Parameter mit Ausnahme der Cache-Konfiguration.
- MODIFY-PUBSET-CACHE-ATTRIBUTES (EDIT-PUBSET-CACHE-ATTRIBUTES)
  Definition der Cache-Konfiguration sowohl für SF-Pubsets als auch für Volume-Sets eines SM-Pubsets.
- REMOVE-MASTER-CATALOG-ENTRY
  Löschen von MRSCAT-Einträgen.

### Zugriffsberechtigung und Benutzerkatalog

Mit Hilfe der Kommandos ADD-USER, MODIFY-USER-ATTRIBUTES, LOCK-USER, REMOVE-USER und UNLOCK-USER legt die Systembetreuung fest, welcher Benutzer auf welchen Pubset zugreifen darf und ob einem Benutzer die Zugriffsberechtigung für einen Pubset entzogen wird. Da jeder Pubset einen eigenen Benutzer- und Dateikatalog hat, kann die Systembetreuung die Benutzerkennungen in Abhängigkeit der Funktionen, die die Benutzer ausüben, auf die Pubsets aufteilen.

Im Benutzerkatalog des Home-Pubsets müssen für sämtliche Benutzer Einträge hinterlegt sein, weil die Zugangsüberprüfungen beim Kommando SET-LOGON-PARAMETERS im Benutzerkatalog des Home-Pubsets durchgeführt werden. Es bietet sich an, die Benutzer in 3 Gruppen einzuteilen:

1. Der Benutzer hat im Benutzerkatalog des jeweiligen Pubsets keinen Eintrag

   Das bedeutet, dass er auf diesen Pubset nicht zugreifen kann. Der Benutzer kann auch auf mehrbenutzbare Dateien oder Jobvariablen anderer auf diesem Pubset geführter Benutzer nicht zugreifen.

   *Beispiel*

   Benutzerkennungen, die lediglich Test- und Ausbildungszwecken dienen, sollten nicht auf Pubsets zugreifen können, die für die Produktion benötigt werden.

2. Der Benutzer hat im Benutzerkatalog des jeweiligen Pubsets einen Eintrag, aber er kann auf diesem Pubset keinen Speicherplatz belegen

   Die Systembetreuung erreicht diesen Fall, der für den Home-Pubset die Regel darstellt, mit dem Operanden PUBLIC-SPACE-LIMIT=0 beim Kommando ADD-USER.
   Dateien, die unter der angegebenen Benutzerkennung bereits existieren, und mehrbenutzbare Dateien anderer Benutzer lassen sich normal verarbeiten.
   Es können aber keine Dateien auf dem Pubset unter dieser Benutzerkennung neu eingerichtet oder erweitert werden.
   Der Zugriff auf Dateien anderer Benutzerkennungen ist den normalen Prüfungen unterworfen (Mehrbenutzbarkeit, Kennwörter, Lese- oder Schreibzugriff usw.). Ein Benutzereintrag mit PUBLIC-SPACE-LIMIT=0 hat keine Auswirkungen auf das Erstellen von Jobvariablen, da sie keinen Speicherplatz auf dem Pubset belegen, sondern nur durch einen Katalogeintrag, der den Namen und den Wert der JV enthält, realisiert werden. Das Erstellen von Jobvariablen auf diesem Pubset kann jedoch durch die Systembetreuung dadurch verhindert werden, dass beim Kommando ADD-USER zusätzlich der Operand JV-NUMBER-LIMIT=0 angegeben wird. Des Weiteren kann durch zusätzliche Angabe des Operanden FILE-NUMBER-LIMIT=0 das Erstellen von Dateieinträgen im Dateikatalog TSOSCAT des betreffenden Pubsets verhindert werden.

   *Beispiel*

   Im Benutzerkatalog des Home-Pubsets sollte für alle Benutzer ein Eintrag mit PUBLIC-SPACE-LIMIT=0, FILE-NUMBER-LIMIT=0 und JV-NUMBER-LIMIT=0 hinterlegt sein. Die Benutzer können ungehindert auf Dateien dieses Pubsets zugreifen, sie dürfen aber keine Dateien erzeugen.

3. Der Benutzer hat im Benutzerkatalog des jeweiligen Pubsets einen Eintrag und kann auf diesem Pubset Speicherplatz belegen

   Für den Benutzer bedeutet dies, dass er ohne besondere Einschränkungen auf dem Pubset Dateien und Jobvariablen erzeugen und verarbeiten kann.
   Einschränkungen ergeben sich für den Benutzer, wenn ihm im Benutzerkatalog zwar Speicherplatz eingeräumt wird, er diesen jedoch wegen der Reglementierung der Datei- oder Jobvariablenanzahl nicht nutzen kann.



Der Zugriffsschutz ist unwirksam, wenn bei der Systemeinleitung der Systemparameter FSHARING den Wert 1 erhalten hat. Dann können alle Benutzer auf sämtliche Pubsets zugreifen und benötigen dazu nicht einmal einen Eintrag im jeweiligen Benutzerkatalog.
Zum Systemparameter FSHARING siehe Seite 744.

*Zuordnen einer Standard-Katalogkennung*

Jedem Benutzer weist die Systembetreuung mit dem Kommando ADD-USER eine Standard-Katalogkennung zu, die den Pubset bestimmt, auf den zugegriffen wird, wenn der Benutzer beim Zugriff auf Dateien oder Jobvariablen keine Katalogkennung angibt.

Die Standard-Katalogkennungen werden im Benutzerkatalog des Home-Pubsets hinterlegt.
Auf dem standardmäßig zugewiesenen Pubset legt die Systembetreuung für die Benutzer das Speicherplatz-Limit fest (Operanden PUBLIC-SPACE-LIMIT und PUBLIC-SPACE-EXCESS der Kommandos ADD-USER und MODIFY-USER-ATTRIBUTES).
Zusätzlich kann auch der Speicherplatz für temporäre Dateien, die Anzahl der Jobvariablen und die Anzahl der Dateieinträge im TSOSCAT dieses Pubsets eingeschränkt werden (Operanden TEMP-SPACE-LIMIT, JV-NUMBER-LIMIT und FILE-NUMBER-LIMIT der Kommandos ADD-USER und MODIFY-USER-ATTRIBUTES).



Wenn die Systembetreuung eine neue Benutzerkennung einführt, die Zugriff zu mehreren Pubsets haben soll, so muss sie für jeden dieser Pubsets ein entsprechendes ADD-USER-Kommando geben. Das gilt analog für das Modifizieren und Löschen von Benutzereinträgen.

## 9.5.2 Benutzer-Kontingente und Ressourcen-Schutz

Die Systembetreuung kann mit Hilfe von Benutzer-Kontingenten den von einzelnen Benutzern auf einem Pubset beanspruchten Platz kontrollieren und einschränken. Dabei wird den Besonderheiten der SM-Pubsets durch eine größere Differenzierung der Benutzer-Kontingente Rechnung getragen.

Zum Festlegen von Speicherplatz-Kontingenten und Überschreitungsbedingungen dienen die Kommandos ADD-USER, MODIFY-USER-ATTRIBUTES und MODIFY-USER-PUBSET-ATTRIBUTES. Letzteres dient zum differenzierten Festlegen der Benutzer-Kontingente von SM-Pubsets, ist aber auch für SF-Pubsets anwendbar.

Sowohl für SF-Pubsets als auch für SM-Pubsets dient zum Anzeigen der eingestellten Speicherplatz-Kontingente das Kommando SHOW-USER-ATTRIBUTES.

#### Kontingentierung auf SF-Pubsets

Auf SF-Pubsets erfolgt eine separate Kontingentierung des von permanenten und temporären Dateien belegten Platzes (PERM-SPACE-Kontingent, TEMP-SPACE-Kontingent). Die Speicherplatz-Kontingentierung beschränkt sich auf den in der Verarbeitungsebene belegten Plattenspeicherplatz. Sie umfasst folgende (pubset-spezifisch wirkende) Funktionen:

- Registrierung des von den einzelnen Benutzern belegten Platzes für permanente und temporäre Dateien (Space-Used-Erfassung).
- Möglichkeiten für den Systembetreuer, für die einzelnen Benutzer obere Schranken für den zur Vefügung stehenden „permanenten“ und „temporären“ Speicherplatz festzulegen (Space-Limits).
- Möglichkeiten für den Systembetreuer, benutzer-spezifisch festzulegen, wie das System beim Versuch der Überschreitung des Space-Limits reagieren soll (Public-Space-Excess). Diese Festlegungen wirken für das PERM-SPACE- und für das TEMP-SPACE-Kontingent identisch.

### Erweiterungen für SM-Pubsets

Für SM-Pubsets wird das Kontingentkonzept durch folgende Punkte erweitert:

- Einführung des Kontingent-Typs „WORK-SPACE“ für den durch Arbeitsdateien belegten Platz.
- Interne Strukturierung der PERM-SPACE-, TEMP-SPACE- und WORK-SPACE-Kontingente in Unterkontingente. Dies ist erforderlich, um die im Verhältnis zu SF-Pubsets komplexere Ressourcen-Situation innerhalb der SM-Pubsets berücksichtigen zu können. Dabei werden auch die Hintergrundebenen für Verdrängung mit einbezogen.

Wie bei SF-Pubsets können benutzer-spezifische Überschreitungsregeln definiert werden. Diese wirken für alle Kontingente bzw. Unterkontingente des SM-Pubsets identisch.

### Kontingentstruktur auf SM-Pubsets für permanente Dateien

Der Kontingentstruktur für permanente Dateien in einem SM-Pubset liegt die Idee zu Grunde, dass der in einem SM-Pubset vorhandene Platz hinsichtlich seiner Qualität durch verschiedene Detaillierungsgrade charakterisiert werden kann. Die Kontingentierung auf SM-Pubsets berücksichtigt die verschiedenen Detaillierungsgrade dadurch, dass einzelne Kontingente Unterkontingente für ihre „höherwertigen“ Anteile enthalten. Der Platz, der durch eine Datei belegt wird, führt zu einer Belastung der Benutzerkontingente. Die betroffenen Benutzerkontingente ergeben sich aus den Dateieigenschaften.

Die einzelnen Kontingente haben folgende Bedeutung:

- TOTAL-SPACE
  Diesem Kontingent wird der Platz für alle auf dem Pubset liegenden permanenten Dateien (eines Benutzers) zugerechnet, unabhängig ob sich diese auf einer Hintergrundebene (S1 oder S2) oder auf der Verarbeitungsebene S0 befinden. Unter Hintergrundebenen sind die durch HSMS genutzten Dateiablageorte zu verstehen. Als Maßeinheit dient hier nicht der real belegte Platz, sondern der Platz, den die Datei nach dem Zurückbringen in die S0-Ebene in dieser belegen würde.
- S0-LEVEL-SPACE
  Gesamtkontingent für permanente Dateien, die sich auf der Verarbeitungsebene S0 befinden.
- HIGH-PERF-SPACE
  Kontingent für Dateien, die einen Performance-Wert *HIGH haben.
- VERY-HIGH-PERF-SPACE
  Kontingent für Dateien, die einen Performance-Wert *VERY-HIGH haben.
- HIGH-AVAILABLE-SPACE
  Kontingent für Dateien, die einen Verfügbarkeitswert *HIGH haben.

In folgenden Bild ist für permanente Dateien der Zusammenhang zwischen Dateiattributen und Kontingenten dargestellt, wobei zu beachten ist, dass die Belastung eines Kontingents implizit auch zu einer Belastung aller übergeordneten Kontingente führt.

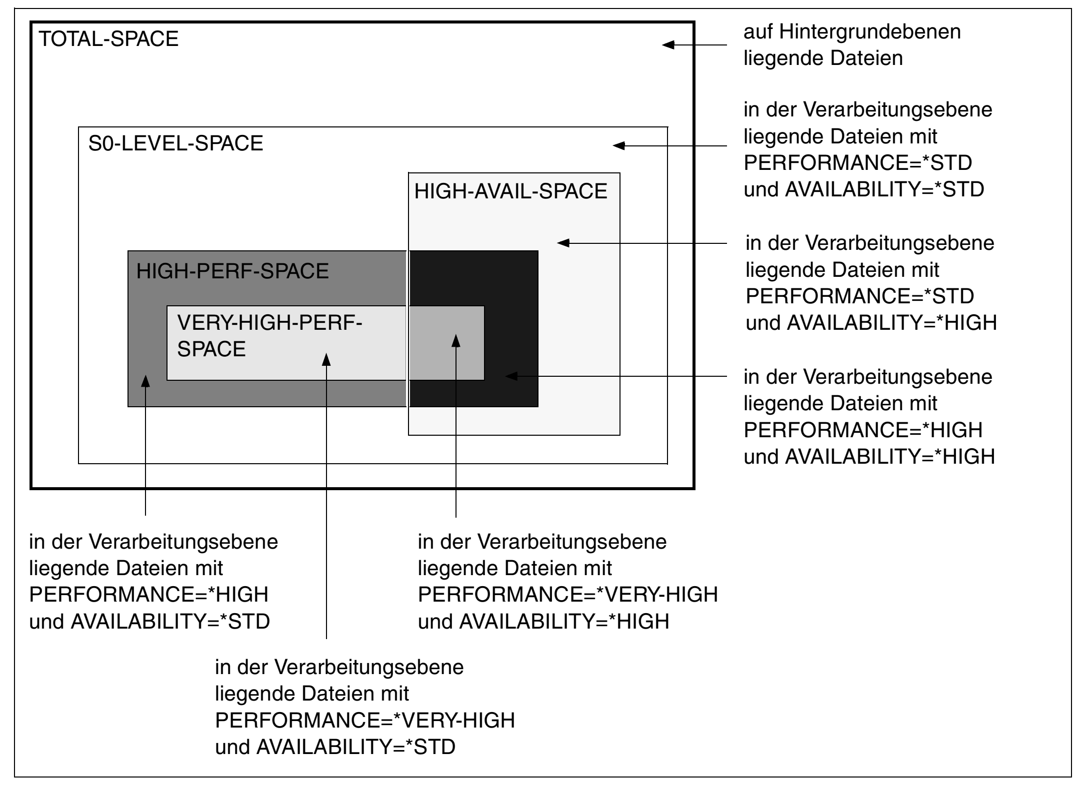



Bild 20: Kontingentstruktur für permanente Dateien in einem SM-Pubset und Zuordnung zu Dateiattributen

*Beispiel zu obigem Bild*

Das Anlegen einer Datei mit den Attributen AVAILABILITY=*HIGH und PERFORMANCE=*HIGH führt dazu, dass neben den Belegt-Werten der Kontingente für HIGH-AVAIL-SPACE und HIGH-PERF-SPACE auch die Belegt-Werte für S0-LEVEL-SPACE und TOTAL-SPACE um die Anzahl von Half-Pages erhöht werden, welche der Datei zugewiesen sind.

### Kontingentstruktur auf SM-Pubsets für temporäre Dateien und Arbeitsdateien

Temporäre Dateien und Arbeitsdateien können nicht auf Hintergrundebenen migriert werden und können nicht das Dateiattibut AVAILABILITY=*HIGH zugewiesen bekommen. Eine Unterscheidung zwischen den Kontingenten für TOTAL-SPACE und S0-LEVEL-SPACE kann damit entfallen, und für das HIGH-AVAILABLE-SPACE-Kontingent besteht kein Bedarf. Daher ergibt sich für temporäre Dateien und Arbeitsdateien im Vergleich zu permanenten Dateien eine vereinfachte Kontingentstruktur, die in folgendem Bild veranschaulicht ist.

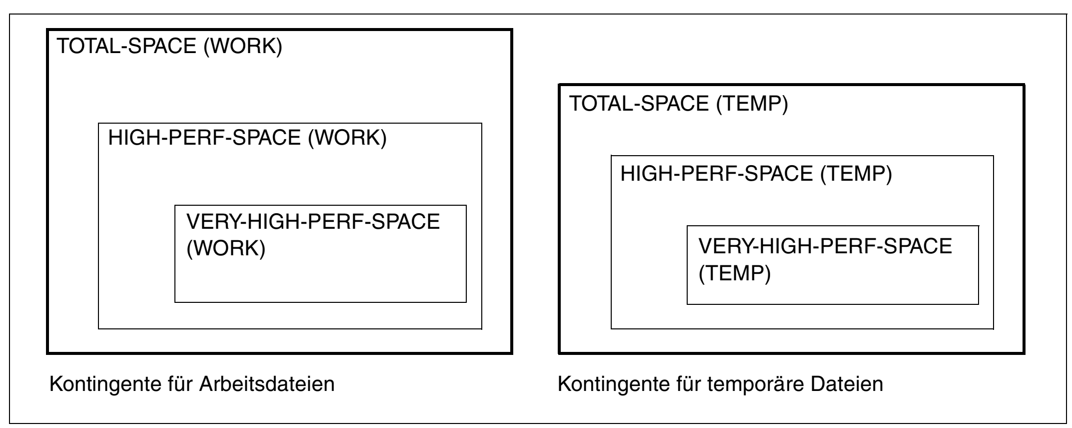



Bild 21: Kontingentstruktur für Arbeitsdateien und temporäre Dateien

Jedem Kontingent ist ein Maximalwert und ein aktueller Belegungswert (Belegt-Wert) zugeordnet. Der aktuelle Belegungswert der Kontingente ergibt sich aus den Dateien eines Benutzers und ihren Dateiattributen. Das System überwacht beim Anlegen und Vergrößern von Dateien, dass die Maximalwerte nicht überschritten werden (sofern keine Sonderberechtigung (PUBLIC-SPACE-EXCESS) vorliegt, welche das Überschreiten des zulässigen Speicherplatzes gestattet).

Zusätzlich zu den Kontingenten gibt es noch weitere Mechanismen für den Ressourcen-Schutz:

### Zähler für Dateien und Jobvariablen

Damit ein Benutzer nicht beliebig viele Dateien oder Jobvariablen auf einem Pubset anlegen kann, wird im Benutzerkatalog pro Benutzer ein Wert für die max. Anzahl von Dateien geführt. Sollte dieser Wert beim Einrichten einer neuen Datei überschritten werden, wird der Auftrag zurückgewiesen.

### Pubset-spezifische Benutzerrechte

Es existieren folgende pubset-spezifische Benutzerrechte:

1. DMS-TUNING-RESOURCES
   Dieses Recht bestimmt, ob es einem Benutzer gestattet ist, Tuning-Ressourcen für die Dateibearbeitung zu nutzen. Tuning-Ressourcen sind die Pubset-Caches und hochwertige Speicherklassen. Das Attribut kann folgende Werte annehmen:
   - NONE:
     Der Benutzer darf keine Tuning-Ressourcen nutzen.
   - CONCURRENT-USE:
     Der Benutzer darf Tuning-Ressourcen nutzen, muss diese jedoch mit anderen Benutzern teilen. Beispiel: Er kann einen Pubset-Cache nutzen, seine eingelagerten Seiten unterliegen jedoch dem Verdrängungsalgorithmus.
   - EXCLUSIVE-USE:
     Der Benutzer darf Tuning-Ressourcen nutzen und braucht diese nicht mit anderen Benutzern zu teilen. Beispiel: Er kann einen Pubset-Cache nutzen, seine eingelagerten Seiten unterliegen keinem Verdrängungsalgorithmus.
2. PHYSICAL-ALLOCATION
   Bestimmt, ob ein Benutzer Plattenspeicherplatz direkt unter Angabe von Volume-Set, Volume oder Block anfordern darf.
3. PUBLIC-SPACE-EXCESS
   Bestimmt, ob ein Benutzer temporär ein Kontingentlimit überschreiten darf.
4. NET-STORAGE-USAGE=<u>*ALLOWED</u>/*NOT-ALLOWED
   Dieses Recht erlaubt oder verbietet dem Benutzer die Nutzung von Net-Storage.
   Der belegte Speicherplatz auf einem Net-Storage-Volume wird nicht auf das Public-Space-Limit des Benutzers angerechnet.

### 9.5.3 In- und Außerbetriebnahme von Pubsets

Das Inbetriebnehmen eines Pubsets wird durch das Kommando IMPORT-PUBSET eingeleitet. Grundvoraussetzung dafür sind die Hardware-Verbindungen zu den jeweiligen Platten.



Ein Pubset, in dem Dateien auf Net-Storage katalogisiert sind, kann auch dann in Betrieb genommen werden, wenn der entsprechende Net-Storage nicht zur Verfügung steht. Es wird aber aus Verfügbarkeitsgründen empfohlen, den Net-Storage zuerst verfügbar zu machen und dann den Pubset in Betrieb zu nehmen, siehe Kapitel „Net-Storage-Verwaltung“ auf Seite 429.

Eine weitere Voraussetzung ist, dass für den Pubset ein MRSCAT-Eintrag existiert, der folgende Informationen enthalten muss:

- Pubset-Kennung
- Pubset-Typ (SF- oder SM-Pubset)
- Kennung des Control-Volume-Sets (nur bei SM-Pubsets)

Alle weiteren Informationen können entweder zum Zeitpunkt der Inbetriebnahme angegeben oder aus den Ablageorten auf dem Pubset selbst ermittelt und in den jeweiligen MRSCAT-Einträgen hinterlegt werden. Die dem Pubset zugeordneten MRSCAT-Einträge müssen während der Pubset-Session erhalten bleiben. Die Definitionen in den Einträgen können zwar modifiziert werden, die Änderungen werden jedoch erst zur nächsten Pubset-Session wirksam oder müssen durch die Funktionen der Pubset-Rekonfiguration aktiviert werden, siehe dazu auch Abschnitt „Dynamische Pubset-Rekonfiguration“ auf Seite 367.

Mit dem Kommando EXPORT-PUBSET kann ein Pubset wieder außer Betrieb genommen werden.

Die In- und Außerbetriebnahme von Daten-Pubsets ist ausschließlich über Kommandoschnittstellen möglich. Der Home-Pubset wird automatisch bei der Systemeinleitung (Startup) in Betrieb bzw. bei der Systembeendigung (Shutdown) außer Betrieb genommen.

Mit dem Kommando SHOW-PUBSET-IMPORT-EXPORT kann sich die Systembetreuung über den Bearbeitungsstand laufender Import- und Export-Aufträge informieren.

#### Statische Pubset-Zustände

Die In- und Außerbetriebnahme eines Pubsets bewirkt jedes Mal einen Wechsel des Pubset-Zustands, der ebenfalls im MRSCAT-Eintrag hinterlegt ist und der Auskunft über die Nutzbarkeit des Pubsets gibt.
Bild 22 auf Seite 361 gibt einen Überblick über die möglichen statischen Pubset-Zustände.

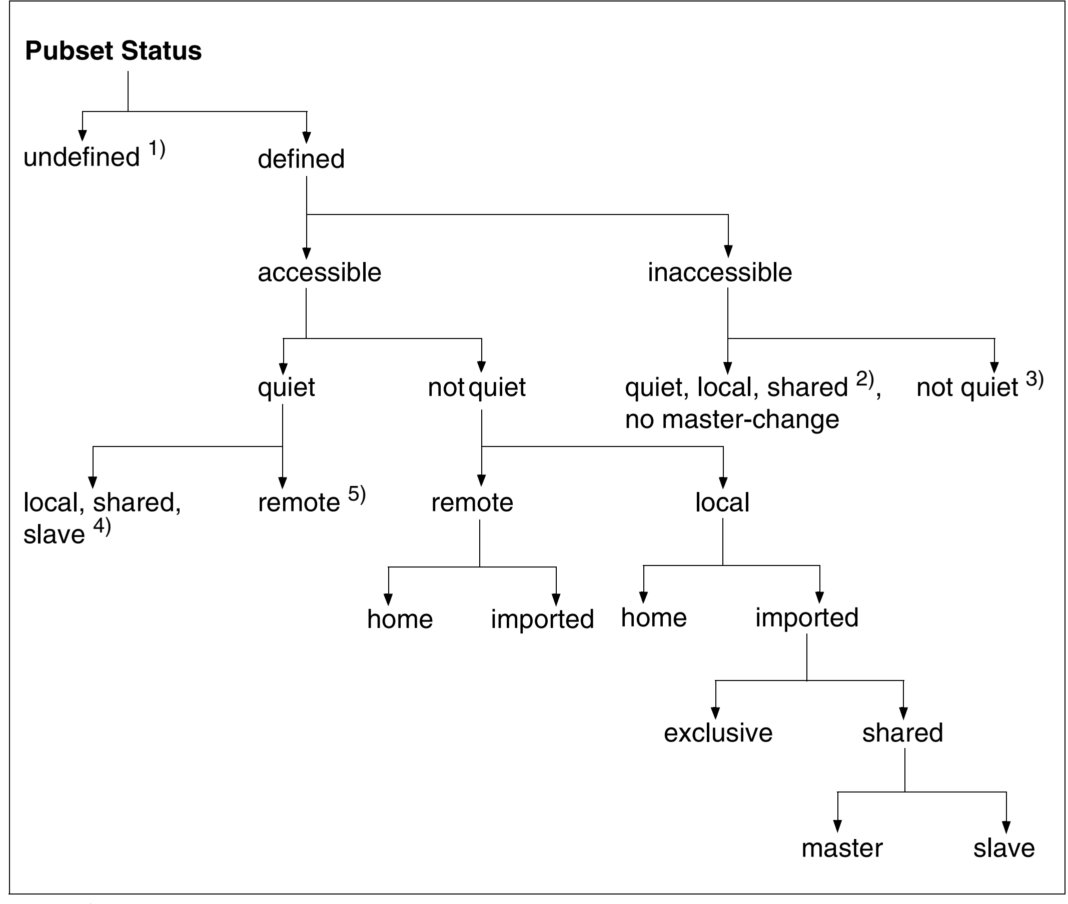



Bild 22: Statische Pubset-Zustände

1) Dieser Zustand bedeutet, dass kein MRSCAT-Eintrag definiert ist.

2) Dieser Zustand wird an einem Pubset-Slave eingenommen, wenn ein Shared Pubset nach Misslingen des Masterwechsels keinen Pubset-Master mehr hat. Der Operator muss explizit mit dem Kommando IMPORT-PUBSET SHARER-TYPE=*MASTER(MASTER-CHANGE=*YES) auf einem der Pubset-Slaves den Master-Wechsel starten.

3) Dieser Zustand wird eingenommen, wenn der Pubset nicht importiert ist.

4) Dieser Zustand wird nach MSCF-Verbindungsausfall beim Pubset-Slave eingenommen.

5) Dieser Zustand wird nach MSCF-Verbindungsausfall eingenommen.

### Besonderheiten der Inbetriebnahme

Im Normalfall werden alle erforderlichen Vorkehrungen automatisch durch das System getroffen und Eingriffe durch den Operator sind i.A. nicht erforderlich. Es können aber nicht vorhersehbare Sondersituationen auftreten, in denen der Import-Vorgang abgebrochen werden muss oder Operatormaßnahmen notwendig sind. Probleme dieser Art sind systeminterne Schwierigkeiten oder Engpässe (z.B. Speicherplatzmangel), Verbindungsprobleme, Volume-Defekte, Mehrdeutigkeit von Volume-Identifizierern, inkonsistente Pubset-Metadaten (z.B. ungültige Zeitstempel), fehlerhafte Current-Master-Zuordnungen, Schwierigkeiten beim Zuordnen der angeforderten Cache-Bereiche, Probleme beim Konnektieren noch zugeordneter Cache-Bereiche, usw.
Im Folgenden sollen einige Problemsituationen genauer betrachtet werden.

*Import-Vorgang gezielt abbrechen*

Bei auftretenden Problemen während des Import-Vorgangs hat die Systembetreuung die Möglichkeit den laufenden Import-Auftrag mit dem Kommando CANCEL-PUBSET-IMPORT vorzeitig abzubrechen. Im Shared-Pubset-Verbund kann insbesondere erreicht werden, dass ein laufender Import mit Master-Wechsel abgebrochen wird und der Pubset zumindest von einem anderen System aus zugreifbar wird.

Bei Aufruf des Kommandos werden dem Import-Auftrag die aktuell bestehenden Plattenzugriffsrechte entzogen und es wird verhindert, dass er neue Plattenzugriffsrechte aufbaut. Der Import-Auftrag selbst kann sich aber nur beenden, wenn er durch den Entzug der Plattenzugriffsrechte auf einen Fehler läuft oder wenn er die Cancel-Anforderung, die im MRSCAT hinterlegt ist, „bemerkt“. Falls sich der Import-Auftrag nicht beendet, kann der Pubset in der laufenden Session nicht mehr importiert werden. Die Pubset-Konsistenz ist jedoch beim Importieren in einem anderen System sichergestellt.

*Homogenitätsprüfung*

Mit Angabe des Operanden CHECK-PUBSET-MIRRORS=*YES im Kommando IMPORT-PUBSET wird für den zu importierenden Pubset die Homogenität der Pubset-Spiegelung überprüft. Sobald bei einem Pubset-Volume eine Inhomogenität festgestellt wird, wird der Import-Vorgang unterbrochen und die Systembetreuung hat die Wahl, Import-Vorgang abzubrechen oder fortzusetzen (siehe Seite 369).

*Probleme bei der Rekonnektierung von Cache-Bereichen*

Bei ordnungsgemäßer Außerbetriebnahme eines Pubsets werden die Zuordnungen zwischen Pubset und Cache-Bereichen aufgehoben, nachdem alle über einen Cache bearbeiteten Dateien auf den Platten in einen konsistenten Zustand gebracht worden sind. Gelingt dies nicht (z.B. bei einem Systemabsturz), bleiben die Cache-Zuordnungen auch für einen exportierten Pubset bestehen. Falls die Systembetreuung in der Zwischenzeit die Cache-Bereiche nicht explizit löscht (Kommando FORCE-DESTROY-CACHE), wird bei der nächsten Inbetriebnahme versucht, die noch zugeordneten Cache-Bereiche zu rekonnektieren.

Auf diese Weise kann die Zerstörung von Dateien verhindert werden, für welche noch Ausgabedaten in dem betroffenen Cache-Bereich zwischengepuffert sind. Scheitert die Rekonnektierung eines Cache-Bereichs, so wird der Import-Vorgang abgebrochen, wenn am Inbetriebnahme-System für den Pubset die Option FORCE-IMPORT=*NO besteht.

Um trotz Fehler bei der Cache-Rekonnektierung die Inbetriebnahme eines Pubsets zu ermöglichen, ist die Option FORCE-IMPORT=*BY-OPERATOR vorgesehen. Sie bewirkt, dass für jeden Cache-Bereich, der nicht rekonnektiert werden kann, der Operator gefragt wird, ob der Import trotzdem (ggf. unter Inkaufnahme von Datenverlusten) fortgesetzt werden soll. Die FORCE-IMPORT-Option kann durch den Systembetreuer mit Hilfe des Kommandos MODIFY-PUBSET-CACHE-ATTRIBUTES passend eingestellt werden.

*Probleme bei Neukonnektierung von Cache-Bereichen*

Die vom Systembetreuer gewünschte Cache-Konfiguration eines Pubsets (Cache-Typ und Cache-Größe) wird durch das Kommando MODIFY-PUBSET-CACHE-ATTRIBUTES festgelegt. Bei SF-Pubsets werden die Festlegungen ausschließlich im MRSCAT-Eintrag des Pubsets vermerkt, bei SM-Pubsets werden sie zusätzlich im Pubset selbst hinterlegt.
Bei SM-Pubsets sind daher im Gegensatz zu den SF-Pubsets Änderungen der gewünschten Cache-Konfiguration nur möglich, wenn der Pubset importiert ist. Während des Pubset-Imports wird vom System versucht, Cache-Bereiche entsprechend den Konfigurationsfestlegungen bereitzustellen. Dabei kann es sich ergeben, dass die gewünschten Cache-Ressourcen nicht oder nicht in ausreichendem Umfang verfügbar sind. Mit Hilfe der Option SIZE-TOLERANCE kann der Systembetreuer steuern, ob in diesem Fall die Inbetriebnahme abgebrochen (SIZE-TOLERANCE=*NO) oder fortgeführt wird (=*YES).
Die SIZE-TOLERANCE-Option kann über das Kommando MODIFY-PUBSET-CACHE-ATTRIBUTES eingestellt werden.

*Probleme als Folge von Plattendefekten*

SF-Pubsets werden immer als Ganzes in Betrieb genommen. Eine partielle Inbetriebnahme, bei welcher einzelne Volumes des Pubsets nicht einbezogen sind, ist nicht möglich.

SM-Pubsets sind so konzipiert, dass die einzelnen Volume-Sets (mit Ausnahme des Control-Volume-Sets) unabhängige Ausfalleinheiten darstellen. Beim Defekt eines Volume-Sets ist es möglich, diesen zu entfernen und den restlichen Pubset regulär weiterzubenutzen. Wenn der SM-Pubset importiert ist, kann dies mit Hilfe der Kommandos der Pubset-Rekonfiguration erfolgen. Für einen exportierten Pubset mit einem defekten Volume-Set sind diese nicht nutzbar. Um auch in diesem Fall den intakten Teil des Pubsets weiter verwenden zu können, bietet das IMPORT-PUBSET-Kommando dem Systembetreuer die Möglichkeit, Volume-Sets (mit Ausnahme des Control-Volume-Sets) anzugeben, die vor der eigentlichen Inbetriebnahme automatisch aus dem Pubset ausgegliedert werden (Operand DEFECT-VOLUME-SET). Die Inbetriebnahme selbst erfolgt nur für den verbleibenden Pubset. Analog der Entfernung eines Volume-Sets in einer laufenden Pubset-Session wird ein Verzeichnis der auf dem ausgegliederten Volume-Set befindlichen Dateien erstellt, das bei einem späteren RESTORE-Aufruf (durch HSMS) verwendet werden kann.

### Pubset-Belegungsfunktion

Voraussetzung für die Außerbetriebnahme eines Pubsets ist, dass kein Benutzer diesen Pubset noch als Ressource verwendet. Die Pubset-Belegungsfunktion stellt fest, ob ein Pubset genutzt wird und von wem.

Bei jeder Anforderung an einen Pubset ruft die Instanz, die diese Anforderung bearbeitet (z.B. der OPEN) die Pubset-Belegungsfunktion auf, um sich als Nutzer des Pubsets registrieren zu lassen. Das Registrieren erfolgt durch Hinterlegen der Aufrufer-Identifikation (TSN). Als Ablageort für die Aufrufer-Identifikation dienen die Belegungseinträge. Belegt eine Task den Pubset erstmalig, wird gleichzeitig ein Belegungszähler in dem dazugehörigen MRSCAT-Eintrag erhöht. Bei einer nochmaligen Belegung derselben Task wird lediglich der Zähler im Belegungseintrag erhöht. Eine Belegungsanforderung wird abgewiesen, wenn der Pubset nicht verfügbar ist oder gerade exportiert wird.

Die Rückgabe der Belegung erfolgt, wenn der Pubset nicht mehr benötigt wird, z.B. beim CLOSE einer Datei. Bei einer Belegungsfreigabe wird der Zähler im Belegungseintrag, bei der letzten Belegungsfreigabe durch eine Task wird der globale Belegungszähler im MRSCAT-Eintrag heruntergezählt. Wenn der globale Belegungszähler im MRSCAT-Eintrag den Wert Null hat, bedeutet das, dass der Pubset nicht mehr genutzt wird und exportiert werden darf.

Das Kommando UNLOCK-DISK bereinigt das Systembelegungsprotokoll (Liste der belegenden Systeme im SVL) einer Platte von Katalogkennungen, die nicht mehr mit dieser Platte arbeiten. Damit kann der Operator noch existierende Dateilocks zurücksetzen. UNLOCK-DISK UNIT=*PUBSET-DEVICES(PUBSET=...) bereinigt das Systembelegungsprotokoll aller Platten des angegebenen Pubsets. Der Operator muss dabei sicherstellen, dass kein System ausgetragen wird, das noch mit der Platte arbeitet.

Auskunft über die Tasks, die momentanen einen Pubset benutzen, gibt das Kommando SHOW-PUBSET-OCCUPATION.

### Besonderheiten der Außerbetriebnahme

Ein Pubset-Export kann erst dann richtig anlaufen, wenn keine Beleger mehr für den Pubset existieren. Solange dies nicht der Fall ist, befindet sich der Exportauftrag in einem Wartezustand. Dieser Wartezustand beendet sich durch Eintreten eines der folgenden Ereignisse:

- Der Exportauftrag wird mit dem Kommando CANCEL-PUBSET-EXPORT abgebrochen. Der Pubset ist dann wieder komplett verfügbar.
- Alle Pubset-Beleger haben ihre Belegung freigegeben, der Pubset-Export beginnt mit der eigentlichen Arbeit.
- Alle Pubset-Beleger werden gezwungen, ihre Belegung freizugeben (Kommando EXPORT-PUBSET ...,TERMINATE-JOBS=*YES). Dadurch werden alle noch belegenden Tasks zwangsweise beendet, wobei in der Terminierungsphase auch die Pubset-Belegungen zurückgegeben werden.
- Letzte Alternative ist das Kommando FORCE-PUBSET-EXPORT. Hier wird trotz noch existierender Belegungen der Pubset-Export eingeleitet. Die Konsequenz ist, dass die Pubset-Verwaltungsdaten nicht korrekt abgebaut werden können, sodass es bei einer erneuten Inbetriebnahme in derselben BS2000-Session zu schweren Inkonsistenzen innerhalb der Verwaltungsdaten kommen und ein reibungsloser Pubset-Betrieb nicht gewährleistet werden kann. Diese Funktion soll also nur als äußerste Notmaßnahme eingesetzt werden.

### Verhalten bei fehlerhaften TSOSCAT-Dateien

Besteht der Verdacht, dass Kontrollstrukturen eines Pubsets (z.B. die TSOSCAT-Datei oder F5-Kennsätze) fehlerhaft sind, kann über den Operanden RECONSTRUCT-F5-LABEL des Kommandos IMPORT-PUBSET eine Rekonstruktion explizit veranlasst werden.

Wenn beim Import eines Pubsets während der Rekonstruktion der F5-Kennsätze in einer TSOSCAT-Datei ein Fehler in der Verkettung der TSOSCAT-Seiten einer Benutzerkennung festgestellt wird (fehlerhafte Benutzerkette), wird im weiteren Verlauf der F5-Rekonstruktion im Katalogmanagement (CMS) ein Analysemodus aktiviert. In diesem Analysemodus wird zur Diagnose die Verkettung der noch zu verarbeitenden Benutzerkennungen an der Konsole protokolliert. Die intakte Verkettung wird jeweils über die Meldung `DMS13A7` (eine oder mehrere pro Kennung) angezeigt. Tritt ein Fehler in einer Benutzerkette auf, wird dies mit der Meldung `DMS13A8` protokolliert.

Diese Information kann dazu verwendet werden, um einen Überblick über das Ausmaß des Schadens an der TSOSCAT-Datei zu erhalten. Der Import des Pubsets wird nach Ende der Analyse in jedem Fall mit Fehler beendet. Ein erfolgreicher Import des Pubsets ist i.A. erst nach Reparatur der betroffenen TSOSCAT-Datei möglich.

In jedem Fall sollten Sie Kontakt mit dem Service aufnehmen. Die Durchführung der Reparatur kann vom Systemspezialisten nach geeigneter Diagnose manuell vorgenommen oder automatisch veranlasst werden.

Die automatische Korrektur kann über den Operanden REPAIR-TSOSCAT=*YES des Kommandos IMPORT-PUBSET aktiviert werden. Während der Import-Phase wird versucht, die beschädigten Benutzerketten durch das Ausketten defekter Blöcke oder das Abhängen des Kettenrests für das Katalogmanagement wieder zugreifbar zu machen.



**ACHTUNG!**
Dateien, deren Katalogeintrag auf den so abgehängten Seiten liegt, gehen verloren! Eine Information, welche Dateien betroffen sind, kann durch das System nicht bereitgestellt werden. In jedem Fall sollte vor Aktivierung der automatischen TSOSCAT-Reparatur der Service kontaktiert werden.

Der Reparatur-Modus muss zur Sicherheit (einmal) über die Meldung `DMS13AE` bestätigt werden. Durchgeführte Maßnahmen werden über die Meldungen `DMS13A9`, `DMS13AA`, `DMS13AB` und `DMS13AC` protokolliert. Ein Fehlschlagen der Reparatur (z.B. auf Grund eines permanenten HW-Fehlers) wird durch die Meldung `DMS13AD` angezeigt.

### 9.5.4 Dynamische Pubset-Rekonfiguration

Die bei der Generierung festgelegte Pubset-Konfiguration kann während der „Lebensdauer“ des Pubsets geändert (rekonfiguriert) werden.
Dazu bestehen mehrere Möglichkeiten:

- Mit Hilfe des Programms SIR zur Pubset-Generierung kann eine leere, zum Pubset oder Volume-Set passende Platte hinzugenommen werden.
- Bei der Inbetriebnahme eines SF-Pubsets wird eine zum Pubset-Namen passend benannte Platte hinzugefügt, die während des Imports in den Pubset eingegliedert wird.
- Über verschiedene Kommandoschnittstellen der Pubset-Rekonfiguration können Konfigurationsdaten sowie die Platten- und Volume-Set-Konfiguration eines Pubset geändert werden (dynamische Pubset-Rekonfiguration).
- Mit SPACEPRO (Space Provisioning) werden Pubsets bezüglich Speicherplatzsättigung überwacht. Wenn bei einem überwachten Pubset eine Speicherplatzsättigung auftritt, analysiert SPACEPRO den Sättigungzustand und erweitert bei Bedarf den Pubset kontrolliert um Volumes aus einem Vorrat freier Volumes (Free-Pool).
- Zur Rekonfiguration von Net-Storage-Volumes siehe Seite 439.

Die beiden erstgenannten Rekonfigurationsvorgänge müssen außerhalb der Betriebsphase des Pubsets erfolgen und bieten zudem lediglich eine Erweiterung der Pubset-Konfiguration.
Der Vorteil der Pubset-Rekonfiguration über die Kommandoschnittstellen ist, dass die Rekonfiguration dynamisch erfolgt, d.h. der Pubset ist in Betrieb.
Außerdem werden auch Funktionen zur Reduktion der Pubset-Konfiguration angeboten.
Die nun folgenden Abschnitte befassen sich ausschließlich mit der funktionalen Beschreibung der dynamischen Pubset-Rekonfiguration.

#### Änderung statischer Konfigurationsdaten (nur bei SM-Pubsets)

Die statischen Konfigurationsdaten eines SM-Pubset werden mit Hilfe des Kommandos MODIFY-PUBSET-DEFINITION-FILE (EDIT-PUBSET-DEFINITION-FILE) verändert. Unter den Begriff „statische Konfigurationsdaten“ eines SM-Pubset fallen die Definition der Volume-Set-Konfiguration und die Service-Attribute der Volume-Sets.
Das Kommando ermöglicht, ein neues Volume-Set mit seinen Eigenschaften für den Pubset zu definieren, die Eigenschaften eines bestehenden Volume-Sets zu ändern oder ein definiertes Volume-Set aus der Pubset-Konfiguration zu löschen (vorausgesetzt, es ist nicht in Betrieb). Die Änderungen der statischen Konfigurationsdaten werden bei der Kommandobearbeitung sowohl in der Pubset-Konfigurationsdatei wie auch im MRSCAT vollzogen. Eine Änderung der dynamischen Konfigurationsdaten, z.B. das In- oder Außerbetriebnehmen eines Volume-Sets, ist mit diesem Kommando nicht möglich.

### Änderung dynamischer Konfigurationsdaten (bei SF- und SM-Pubsets)

Das Kommando MODIFY-PUBSET-PROCESSING erlaubt das Ändern der dynamischen Konfigurationsdaten bezogen auf die Volume-Set-Konfiguration eines SM-Pubsets und die Plattenkonfiguration von Volume-Sets oder SF-Pubsets:

- Volume-Set-Rekonfiguration

  Die Abläufe bei der Volume-Set-Rekonfiguration sind mit den Abläufen der In- und Außerbetriebnahme von Volume-Sets beim Pubset-Import bzw. Export vergleichbar. Unterschiede bestehen lediglich bei der Überprüfung bestimmter Voraussetzungen für den jeweiligen Rekonfigurationsauftrag.

  Bevor ein Volume-Set für ein SM-Pubset aktiviert, d.h. in Betrieb genommen werden kann, muss es mit dem Kommando MODIFY-PUBSET-DEFINITION-FILE (EDIT-PUBSET-DEFINITION-FILE) definiert worden sein.

  Voraussetzung für das Deaktivieren eines Volume-Sets ist, dass es keine Dateien mehr beinhaltet und der betroffene Volume-Set gegen Primär-Allokierungen (siehe Seite 371) gesperrt ist.
  Eine Ausnahme hiervon ist ein defekter Volume-Set, der zwangsdeaktiviert werden kann, wodurch alle Dateien, die sich auf ihm befanden, verloren sind. Es wird deshalb beim Deaktivieren eines defekten Volume-Sets eine Liste aller auf dem Volume-Set befindlichen Dateien erstellt, die direkt als Eingabedatei für Restaurierungsaufträge an das Subsystem HSMS genutzt werden kann. Die Liste wird in der Datei $TSOS.SYS.PUBSET.DEFECT.<volume-set-id>.<datum.uhrzeit> abgelegt.
  Der Defektzustand eines Volume-Sets wird entweder durch das System selbst erkannt, z.B. wenn der Metadatenzugriff auf ein Volume-Set nicht mehr möglich ist, oder kann durch das Kommando MODIFY-PUBSET-RESTRICTION eingestellt werden.

- Platten-Rekonfiguration

  Mit Hilfe von MODIFY-PUBSET-PROCESSING ist es möglich, einzelne Platten aus einem Volume-Set bzw. SF-Pubset zu entfernen oder einen Volume-Set bzw. SF-Pubset um einzelne Platten zu erweitern.
  Bei der Entfernung von Platten wird zunächst geprüft, ob die Platte leer ist, anschließend werden die Datenstrukturen bzgl. der Platte bereinigt und die Zugriffsrechte freigegeben. Bei der Hinzunahme einer leeren Platte müssen die Platteneigenschaften mit den Eigenschaften des Volume-Sets bzw. SF-Pubsets verträglich sein.

  Mit den Kommandos ATTACH-/DETACH-DEVICE UNIT=*PUBSET-DEVICES(...) können Platten eines Pubsets zu- bzw. weggeschaltet werden. Da die Namen der zugehörigen Platten im SVL der Systemplatte verwaltet werden, muss der Pubset mindestens einmal importiert worden sein. Der Eintrag erfolgt jeweils beim Importieren bzw. Exportieren des Pubset sowie bei Änderungen der Pubset-Zusammensetzung (MODIFY-PUBSET-PROCESSING).

Es werden stets so viele Platten wie möglich zugeschaltet. Bei Spiegelung mit DRV werden jeweils beide Platten zugeschaltet. Bei Spiegelung in einer Plattenspeichersystem-Steuerung werden nur die Standardplatten zugeschaltet. Sollen die Spiegelplatten zugeschaltet werden, muss im Operanden PUBSET die Spiegelplatte der Systemplatte (Pubres) angegeben werden.

Mit dem Kommando SHOW-DEVICE-CONFIGURATION UNIT=*PUBSET-DEVICES(...) können Informationen über die zum Pubset gehörenden Platten ausgegeben werden.

Pubres bzw. Volres können nicht aus dem SF-Pubset bzw. Volume-Set entfernt werden.

### Homogenitätsprüfung

Bei der Pubset-Erweiterung mit dem Kommando MODIFY-PUBSET-PROCESSING wird bei Angabe des Operanden CHECK-PUBSET-MIRRORS=*YES eine Homogenitätsprüfung zur Spiegelung durchgeführt: Wenn im Verlauf der Pubset-Erweiterung ein Volume ermittelt wird, das unterschiedliche Spiegelungs-Eigenschaften im Vergleich zu bereits bearbeiteten Volumes aufweist, wird die beantwortbare Meldung `DMS1369` auf SYSOUT ausgegeben.

Abhängig von der Antwort des Aufrufers wird eine der folgenden Vorgehensweisen gewählt:

- Das Erweitern des Pubsets wird abgebrochen.
- Das Erweitern des Pubsets wird trotz festgestellter Inhomogenität für das gerade bearbeitete Volume des Pubsets fortgesetzt. Dabei wird für jedes weitere Volume mit unterschiedlichen Spiegelungs-Eigenschaften die Meldung `DMS136B` an der Bedienstation ausgegeben.

Zur Homogenitätsprüfung muss das Subsystem SHC-OSD verfügbar sein.

Bei der autonomen Pubset-Erweiterung mit SPACEPRO kann die Homogenitätsprüfung über den gleichnamigen Operanden im Kommando MOVE-SPACEPRO-DISK bzw. in den SPACEPRO-Optionen des Pubsets eingestellt werden (siehe Seite 380).

Mit dem Kommando CHECK-PUBSET-MIRRORS kann die Homogenität der Pubset-Spiegelung auch vor der dynamischen Änderung der Pubset-Konfiguration geprüft werden.

Bei Inbetriebnahme eines Pubsets kann die Homogenitätsprüfung im gleichnamigen Operanden des Kommandos IMPORT-PUBSET angefordert werden.

*Kriterien für die Homogenität der Pubset-Spiegelung*

Ein Pubset ist homogen bezüglich lokaler Spiegelung, wenn folgende Bedingungen erfüllt sind:

a) Allen Volumes des Pubsets sind gleich viele Spiegelplatten zugeordnet.

b) Alle Spiegelplatten des Pubsets befinden sich im gleichen Betriebszustand. Im Wesentlichen sind hier nur die Betriebszustände `ESTABLISHED` und `SPLIT` von Bedeutung; die Zustände `ESTABLISHING` und `SPLITTING` treten nur temporär als Zwischenzustände auf.

c) Wenn zusätzlich mit entfernten Spiegelplatten gearbeitet wird, so befinden sich alle dem Pubset zugeordneten Spiegelplatten entweder in ihrer Gesamtheit im lokalen oder im entfernten Plattenspeichersystem, oder es gibt sowohl im lokalen wie auch im entferntern Plattenspeichersystem einen vollständigen Satz von Spiegelplatten.

Bezüglich der entfernten Spiegelung ist der Pubset homogen, wenn folgende Bedingungen erfüllt sind:

a) Allen Volumes des Pubsets sind im entfernten Plattenspeichersystem gleich viele Spiegelplatten zugeordnet.

b) Alle entfernten Spiegelplatten sind im gleichen Zustand `ACTIVE` oder `IN-HOLD`.

c) Alle entfernten Spiegelplatten werden im gleichen Modus betrieben (synchronous, semi-synchronous oder adaptive).

d) Alle entfernten Spiegelplatten befinden sich im selben entfernten Plattenspeichersystem.

e) Für Symmetrix-Plattenspeichersysteme gilt:
Für alle SRDF-Spiegel ist der DOMINO-Modus gleich eingestellt (Operand ON-ERROR im Kommando MODIFY-REMOTE-COPY-PARAMETERS).

### Zugangs- und Allokierungseinschränkungen

Voraussetzung für eine Reduktion der Pubset-Konfiguration ist immer, dass das herauszunehmende Objekt keine Daten mehr beinhaltet. Ein wesentliches Hilfsmittel zur Erfüllung dieser Voraussetzung ist das Kommando MODIFY-PUBSET-RESTRICTIONS.
Der Systembetreuer kann damit für das jeweilige Objekt Zugangs- und Allokierungseinschränkungen vereinbaren bzw. freigeben.

- Allokierungseinschränkungen

  Sie können für Platten oder Volume-Sets vergeben werden. Dabei können Platten gänzlich gegen Allokierungen gesperrt oder nur für physikalische Allokierung freigegeben werden. Für Volume-Sets besteht die Möglichkeit, sie gegen Primär-Allokierungen zu sperren oder sie nur für physikalische Allokierung freizugeben; wobei unter physikalischer Allokierung die Direktangabe der Volume-Set-Kennung zu verstehen ist.

- Zugangssperren

  Sie existieren ausschließlich für Volume-Sets. Durch das Setzen einer Zugangssperre bleibt jeglicher Zugriff auf eine auf dem Volume-Set befindliche Datei dem Systembetreuer vorbehalten. Zugleich kann ein Volume-Set vollständig für die Pubset-Session stillgelegt werden, d.h., jeglicher Zugriff auf den Volume-Set, auch auf die darauf befindlichen Metadaten, ist verboten. Diese Zugangssperre empfiehlt sich dann, wenn ein Defekt, z.B. durch fehlgeschlagene I/O-Aufträge, für den Volume-Set abzusehen ist. Kann diese Zugangssperre nicht mehr aufgehoben werden, gilt der Volume-Set anschließend als defekt.

### Modifikation der Cache-Konfiguration

Das Kommando MODIFY-PUBSET-CACHE-ATTRIBUTES (EDIT-PUBSET-CACHE-ATTRIBUTES) erlaubt eine Änderung der Cache-Konfiguration (Cache-Typ, Cache-Größe) und der Cache-Betriebsparameter. Die Änderungen kommen jedoch erst dann zum Tragen, wenn der SF-Pubset bzw. Volume-Set mit einem Cache-Bereich konnektiert wird, also entweder beim nächsten IMPORT-PUBSET des gesamten Pubsets oder – falls während der Pubset-Session dem betroffenen Volume-Set oder SF-Pubset noch kein Cache-Bereich zugeordnet war – mit Hilfe des Kommandos START-PUBSET-CACHING.

Das Kommando STOP-PUBSET-CACHING erlaubt es, während der Pubset-Session einen konnektierten Cache-Bereich von einem SF-Pubset oder Volume-Set zu diskonnektieren.

### 9.5.5 Pubset-Locks

Die Locks des Pubset-Managements (Pubset-Locks) dienen im Shared-Pubset-Verbund dazu, Pubset-Rekonfigurationsaufträge (siehe Seite 367) untereinander und diese mit Import- und Export-Aufträgen (siehe Seite 360) zu synchronisieren. Pubset-Locks werden auch verwendet, wenn verhindert werden soll, dass sich an der Volume-Konfiguration eines Pubsets etwas ändert, z.B. beim Anlegen eines Snapsets.

Zur besseren Diagnose von Lock-Zuständen und zur Behebung von fehlerhaften Lockzuständen gibt es die Kommandos SHOW-PUBSET-LOCKS und REMOVE-PUBSET-LOCK (Privileg TSOS), siehe Handbuch „Kommandos“ [27].

Da es sich bei Pubset-Locks um interne Sperren des Pubset-Managements handelt, sollte eine manuelle Freigabe nur in abnormalen Lock-Situationen erforderlich sein, z.B. nach dem Ausfall einer MSCF-Verbindung im Shared-Pubset-Verbund. Eine manuelle Lock-Freigabe durch das Kommando REMOVE-PUBSET-LOCK wird nur dann zugelassen, wenn das Pubset-Management über interne Schnittstellen feststellen kann, dass der Task des Lock-Halters nicht mehr lebt oder sich im Zustand „pended indefinitely“ befindet.

#### Typen von Pubset-Locks

Jeder Pubset-Lock wird in Form eines Lock-Eintrags entweder am Pubset-Master oder an einem der Pubset-Slaves (siehe Seite 398) geführt. Das System, an dem der Lock-Eintrag des Locks geführt wird, wird auch als Lock-Location bezeichnet. Zum Lock-Eintrag gehören Informationen über den Typ und den Halter des Pubset-Locks (Task-Id und System-Id).

Es gibt folgende Typen von Pubset-Locks:

- PUBSET-RECONFIGURATION
  Dient dem Schutz von Pubset-Rekonfigurationsaufträgen untereinander und gegen Import- und Export-Aufträge.
- SHARED-EXCAT (werden von Export-Tasks gehalten)
  Verhindert Pubset-Rekonfigurationsaufträge, erlaubt aber parallele Pubset-Locks.
- SHARED-IMCAT (werden von Import-Tasks gehalten)
  Verhindert Pubset-Rekonfigurationsaufträge, erlaubt aber parallele Pubset-Locks.
- SHARED-MASTER-EXCAT
  Verhindert Pubset-Rekonfigurationsaufträge, erlaubt aber parallele Pubset-Locks des Typs SHARED-EXCAT

### Lock-Zustände am Pubset-Master

- PUBSET-RECONFIGURATION
  Wenn dieser Pubset-Lock am Pubset-Master gesetzt ist, dann werden alle nachfolgenden Pubset-Lock-Anforderungen am Pubset-Master abgewiesen. Lediglich Tasks, die einen Pubset-Lock vom Typ PUBSET-RECONFIGURATION anfordern, werden in einen Wartezustand versetzt.

- SHARED-IMCAT
  Wenn dieser Pubset-Lock am Pubset-Master gesetzt ist, dann werden alle nachfolgenden Pubset-Lock-Anforderungen vom Typ PUBSET-RECONFIGURATION am Pubset-Master abgewiesen.

  Tasks, die am Pubset-Master einen Pubset-Lock vom Typ SHARED-MASTER-EXCAT anfordern, werden am Pubset-Master in einen Wartezustand versetzt.

  Erlaubt sind parallele Pubset-Locks vom Typ SHARED-IMCAT (angefordert von Slave-Import-Aufträgen) und SHARED-EXCAT (angefordert von Slave-Export-Aufträgen).

- SHARED-EXCAT
  Wenn einer dieser Pubset-Locks am Pubset-Master gesetzt ist, dann werden alle nachfolgenden Lock-Anforderungen vom Typ PUBSET-RECONFIGURATION am Pubset-Master abgewiesen.

  Erlaubt sind parallele Pubset-Locks der Typen
  – SHARED-IMCAT (angefordert von Slave-Import-Aufträgen)
  – SHARED-EXCAT (angefordert von Slave-Export-Aufträgen) und
  – SHARED-MASTER-EXCAT (angefordert von Master-Export-Aufträgen)

  Tasks, die am Pubset-Slave einen Pubset-Lock vom Typ SHARED-EXCAT oder SHARED-IMCAT anfordern und abgewiesen werden, werden am Pubset-Slave in einen Wartezustand versetzt.

- SHARED-MASTER-EXCAT
  Wenn dieser Pubset-Lock am Pubset-Master gesetzt ist, dann werden alle nachfolgenden Lock-Anforderungen der Typen PUBSET-RECONFIGURATION und SHARED-IMCAT am Pubset-Master abgewiesen. Lediglich Tasks, die einen Pubset-Lock vom Typ SHARED-MASTER-EXCAT (nur am Pubset-Master) anfordern, werden in einen Wartezustand versetzt.

  Erlaubt sind parallele Pubset-Locks des Typ SHARED-EXCAT (angefordert von Slave-Export-Aufträgen).

| angeforderter Pubset-Lock | vorhandener Pubset-Lock | | | |
|---|---|---|---|---|
| | **PUBSET-RECON-FIGURATION** | **SHARED-IMCAT** | **SHARED-EXCAT** | **SHARED-MASTER-EXCAT** |
| **PUBSET-RECON-FIGURATION** | Task wartet | nicht zulässig | nicht zulässig | nicht zulässig |
| **SHARED-IMCAT** | nicht zulässig | zulässig | zulässig | nicht zulässig |
| **SHARED-EXCAT** | nicht zulässig | zulässig | zulässig | zulässig |
| **SHARED-MASTER-EXCAT** | nicht zulässig | Task wartet | zulässig | Task wartet |

Tabelle 24: Kombinationen von Pubset-Locks am Pubset-Master

### Lock-Zustände am Pubset-Slave

Am Pubset-Slave kann nur ein Lock vom Typ PUBSET-RECONFIGURATION oder SHARED-EXCAT gesetzt sein. Pubset-Locks der Typen SHARED-IMCAT und SHARED-MASTER-EXCAT kommen am Pubset-Slave nicht vor.

- PUBSET-RECONFIGURATION
  Wenn dieser Pubset-Lock am Pubset-Slave gesetzt ist, dann werden alle nachfolgenden Lock-Anforderungen am Pubset-Slave abgewiesen. Lediglich Tasks, die einen Pubset-Lock vom Typ PUBSET-RECONFIGURATION anfordern, warten bis zu einer Minute auf die Lock-Freigabe. Nach Ablauf dieser Zeit beenden sich diese Tasks und geben eine entsprechende Fehlermeldung aus.

- SHARED-EXCAT
  Wenn dieser Pubset-Lock am Pubset-Slave gesetzt ist, dann werden alle nachfolgenden Lock-Anforderungen am Pubset-Slave abgewiesen.

  Tasks, die am Pubset-Slave einen Pubset-Lock vom Typ SHARED-EXCAT anfordern, warten bis zu einer Minute auf die Lock-Freigabe. Nach Ablauf dieser Zeit beenden sich diese Tasks und geben eine entsprechende Fehlermeldung aus.

### Beispiele für normale Lock-Situationen

In den folgenden Beispielen besteht der Shared-Pubset-Verbund aus dem Pubset-Master mit der SysId „183“ und zwei Pubset-Slaves mit den SysIds „184“ und „185“.

1. Folgende Lock-Situation kann z.B. während der Bearbeitung von /MODIFY-PUBSET-PROCESSING auftreten. Das Kommando wurde am Pubset-Master oder am Pubset-Slave (in diesem Fall wird das ganze Kommando an den Pubset-Master geschickt) eingegeben. Unter dem Schutz eines PUBSET-RECONFIGURATION-Locks am Pubset-Master werden PUBSET-RECONFIGURATION-Locks an den Pubset-Slaves gesetzt und nach Beendigung der dort durchzuführenden Verarbeitung wieder freigegeben.

```
/show-pubset-locks pubset-id=puba
LOCK-TYPE      LOCK-LOCATION                    LOCK-HOLDER-INFORMATION
               HOSTNAME  SYSID  SHARER-    TID       SYSID  BS2000
                                TYPE                        Version
*PUBSET-RECONF D017ZE15  183    *MASTER    1000004F  183    V18.0
*PUBSET-RECONF D017ZE16  184    *SLAVE     2000007F  184    V18.0
*PUBSET-RECONF D017ZE17  185    *SLAVE     3000009A  185    V18.0
```

   2000007F ist die TID eines MSCF-Servertasks am Pubset-Slave „184“;
   3000009A ist die TID eines MSCF-Servertasks am Pubset-Slave „185“

   *Anmerkung*
   Nicht zu jedem Zeitpunkt müssen die PUBSET-RECONFIGURATION-Locks am Pubset-Slave gesetzt sein. Abhängig vom Stand der jeweiligen Verarbeitung können diese Locks noch nicht gesetzt oder bereits wieder freigegeben sein.

2. Folgende Lock-Situation kann während der Bearbeitung von /EXPORT-PUBSET, das am Pubset-Slave „184“ eingegeben wurde, auftreten.

```
/show-pubset-locks pubset-id=puba
LOCK-TYPE      LOCK-LOCATION                    LOCK-HOLDER-INFORMATION
               HOSTNAME  SYSID  SHARER-    TID       SYSID  BS2000
                                TYPE                        Version
*SHARED-EXCAT  D017ZE16  184    *SLAVE     2000007E  184    V18.0
*SHARED-EXCAT  D017ZE15  183    *MASTER    2000007F  184    V18.0
```

   2000007F ist die TID eines Export-Tasks am Pubset-Slave „184“.

   *Anmerkung*
   Nicht zu jedem Zeitpunkt müssen die SHARED-EXCAT-Locks an Pubset-Slave und Pubset-Master gleichzeitig gesetzt sein. Abhängig vom Stand der Verarbeitung kann der SHARED-EXCAT-Lock auch nur an einem der beiden Systeme gesetzt sein.
   Am Pubset-Slave „185“ ist in diesem Beispiel kein SHARED-EXCAT-Lock gesetzt, da dieses System vom Export-Auftrag an Pubset-Slave „184“ nicht betroffen ist.

### Abnormale Lock-Situationen für ein Pubset im Shared-Pubset-Verbund

Die Anzeigen des Kommandos SHOW-PUBSET-LOCKS sehen in abnormalen Situationen genauso aus wie im obigen Abschnitt „Beispiele für normale Lock-Situationen“. Für die Beurteilung der Lock-Situation ist es wichtig, den Zustand der Tasks der Lock-Halter zu beurteilen. Diese Tasks sind für die Freigabe des entsprechenden Pubset-Locks zuständig.

*PUBSET-RECONFIGURATION-Lock am Pubset-Master gesetzt*

- Wenn die Task, die der Lock-Halter ist, nicht mehr existiert, dann wird der Lock durch eine periodische Überwachungsfunktion nach einer gewissen Zeit automatisch entfernt.
- Wenn die Task, die der Lock-Halter ist, den Status „pended indefinitely“ hat, dann liegt eine abnormale Lock-Situation vor. Normalerweise kann diese Task den Pubset-Lock nicht mehr selbst freigeben.

*SHARED-EXCAT-Lock oder SHARED-IMCAT-Lock am Pubset-Master gesetzt*

- Wenn zweifelsfrei festgestellt werden kann, dass die Task des Lock-Halters (der sich an einem Pubset-Slave befindet) nicht mehr existiert, dann wird der Pubset-Lock durch eine periodische Überwachungsfunktion nach einer gewissen Zeit automatisch entfernt.

  Wenn sich dies nicht feststellen lässt (z.B. wegen MSCF-Verbindungsausfall), dann liegt eine abnormale Lock-Situation vor, wenn auf einem anderen Weg festgestellt werden kann, dass die Task des Lock-Halters nicht mehr existiert (z.B. könnte die Systembetreuung direkten Zugriff auf den Pubset-Slave haben und sich dort Informationen über alle am Pubset-Slave lebenden Tasks geben lassen). Normalerweise kann diese Task den Pubset-Lock dann nicht mehr selbst freigeben. Wenn es sich lediglich um den Ausfall der MSCF-Verbindung zum Pubset-Slave handelt, dann wird die Lock-Situation nach Wiederherstellung der MSCF-Verbindung automatisch durch den periodischen Überwachungsmechanismus bereinigt.
- Wenn am Pubset-Slave die Task des Lock-Halters noch existiert, dann wird gerade ein Export oder Import am Pubset-Slave durch diese Task bearbeitet. Es liegt nur dann eine abnormale Lock-Situation vor, wenn die Task, die der Lock-Halter ist, den Status „pended indefinitely“ hat. Normalerweise kann diese Task den Pubset-Lock dann nicht mehr selbst freigeben.

*SHARED-MASTER-EXCAT-Lock am Pubset-Master gesetzt*

- Wenn das Pubset am Pubset-Master exportiert ist, dann ist der vorhandene Pubset-Lock unschädlich und wird beim nächsten Import des Pubsets am Pubset-Master automatisch entfernt.
- Wenn die Task des Lock-Halters nicht mehr existiert, dann wird der Pubset-Lock durch eine periodische Überwachungsfunktion nach einer gewissen Zeit automatisch entfernt.
- Wenn die Task des Lock-Halters existiert und das Pubset am Pubset-Master noch nicht exportiert ist, dann liegt eine abnormale Lock-Situation vor, wenn die Task, die der Lock-Halter ist, den Status „pended indefinitely“ hat. Normalerweise kann diese Task den Pubset-Lock dann nicht mehr selbst freigeben.
- Vor dem Aufruf zur Freigabe des SHARED-MASTER-EXCAT-Lock durch das Pubset-Master-Export-Processing wartet das System solange bis auf allen Pubset-Slaves das Pubset exportiert ist. Wenn der Pubset-Status am Pubset-Master nicht wieder auf „importiert“ wechselt, dann ist diese Situation unschädlich, da auf dem Pubset ohnehin nicht mehr gearbeitet werden kann. Andernfalls liegt eine abnormale Lock-Situation vor (schwerer Systemfehler).

*PUBSET-RECONFIGURATION-Lock am Pubset-Slave gesetzt*

- Wenn das Pubset am Pubset-Slave exportiert ist, dann ist der vorhandene Pubset-Lock unschädlich und wird beim nächsten Import des Pubsets am selben System automatisch entfernt.
- Wenn das Pubset am Pubset-Slave importiert ist und wenn die Task, die der Lock-Halter ist, nicht mehr existiert, dann liegt eine abnormale Lock-Situation vor.
- Wenn das Pubset am Pubset-Slave importiert ist und die Task, die der Lock-Halter ist, noch existiert, dann liegt dann eine abnormale Lock-Situation vor, wenn die Task, die der Lock-Halter ist, den Status „pended indefinitely“ hat. Normalerweise kann diese Task den Pubset-Lock dann nicht mehr selbst freigeben.

*SHARED-EXCAT-Lock am Pubset-Slave gesetzt*

- Wenn das Pubset am Pubset-Slave exportiert ist, dann ist der Pubset-Lock unschädlich und wird beim nächsten Import des Pubsets am selben System automatisch entfernt. Der Lock-Zustand ist auch dann bedeutungslos, wenn die Task, die der Lock-Halter ist, nicht mehr existiert.
- Wenn das Pubset am Pubset-Slave importiert ist und die Task, die der Lock-Halter ist, noch existiert, dann wird das Pubset gerade am Pubset-Slave exportiert. Es liegt nur dann eine abnormale Lock-Situation vor, wenn die Task, die der Lock-Halter ist, den Status „pended indefinitely“ hat. Normalerweise kann diese Task den Pubset-Lock dann nicht mehr selbst freigeben.

### 9.5.6 SPACEPRO: Autonome Pubset-Rekonfiguration

Das Subsystem SPACEPRO (Space Provisioning) überwacht Pubsets bezüglich Speicherplatzsättigung und erweitert sie bei Bedarf autonom.
Wenn bei einem überwachten Pubset eine Speicherplatzsättigung (siehe auch Seite 393) auftritt, analysiert SPACEPRO den Sättigungzustand und erweitert bei Bedarf den Pubset kontrolliert um Volumes aus einem Vorrat freier Volumes (Free-Pool).
Die automatisierte Anpassung der Anzahl der Platten eines Pubsets vermeidet Bedienungsfehler und ermöglicht einen (vorübergehend) unbedienten Betrieb.

Die Überwachung und automatische Erweiterung mit SPACEPRO ist unabhängig vom Pubset-Typ (SM oder SF) für alle „normal“ genutzten, d.h. System- und Daten-Pubsets eines BS2000-Systems möglich. Das BS2000-System muss den Pubset entweder lokal exklusiv oder als Pubset-Master importiert haben. Ein mit SPACEPRO erweiterbarer Pubset wird auch **SPACEPRO-Pubset** genannt.

Die Pubset-Überwachung übernimmt der **SPACEPRO-Monitor**. Im Wesentlichen wertet der Monitor über PROP-XT die Meldungen über erreichte Sättigungsstufen für die überwachten Pubsets aus und löst ggf. automatisch eine Pubset-Erweiterung aus.

Die Reduzierung eines Pubsets als SPACEPRO-Aktion bei Unterschreiten einer bestimmten Pubset-Belegung wird vom SPACEPRO-Monitor nicht automatisch ausgelöst, kann aber manuell über das Kommando MOVE-SPACEPRO-DISK angestoßen werden.

Statt über den SPACEPRO-Monitor kann die Pubset-Überwachung auch über die Komponente INSPECTOR des openSM2 erfolgen. In diesem Fall werden die SPACEPRO-Aktionen (sowohl Erweiterung als auch Reduzierung) abhängig von frei definierten Pubset-Füllungsgraden automatisch ausgelöst (siehe Abschnitt „Pubset-Überwachung mit INSPECTOR“ auf Seite 390).

Der **Free-Pool** besteht aus einem oder mehreren **Pool-Pubsets** (Pubset-Typ SM oder SF). Ein Pool-Pubset darf nicht gleichzeitig ein SPACEPRO-Pubset sein, also kein „normal“ genutzter Pubset. Der Pool-Pubset dient ausschließlich der Bereitstellung freier Volumes für die autonome Pubset-Erweiterung. Damit diese Volumes ohne Probleme und Aufwand daraus entnommen werden können, müssen sie mit einer Allokierungssperre gegen versehentliche Datenablage geschützt werden. Für Pubres bzw. Volres des Control-Volume-Sets ist keine Allokierungssperre notwendig, da diese nicht für Pubset-Erweiterungen verwendet werden.

| Kommando | Funktion |
| --- | --- |
| MODIFY-PUBSET-SPACEPRO-OPTIONS | SPACEPRO-Optionen ändern |
| MODIFY-SPACEPRO-PARAMETERS | SPACEPRO-Parameter ändern |
| MOVE-SPACEPRO-DISK | SPACEPRO- oder Pool-Pubset erweitern oder reduzieren |
| SHOW-PUBSET-SPACEPRO-HISTORY | SPACEPRO-Historie eines Pubsets anzeigen |
| SHOW-PUBSET-SPACEPRO-OPTIONS | SPACEPRO-Optionen eines Pubsets anzeigen |
| SHOW-SPACEPRO-PARAMETERS | SPACEPRO-Parameter anzeigen |
| SHOW-SPACEPRO-STATUS | Status des SPACEPRO-Monitors anzeigen |
| SIMULATE-SPACEPRO-EVENT | SPACEPRO-Konfiguration testen |
| START-SPACEPRO-MONITORING | SPACEPRO-Monitor für Pubset-Überwachung starten |
| STOP-SPACEPRO-MONITORING | SPACEPRO-Monitor beenden |

Tabelle 25: Übersicht der SPACEPRO-Kommandos

Die Kommandos sind im Handbuch „Kommandos“ [27] beschrieben.

#### Einsatz unter VM2000

SPACEPRO kann unter VM2000 in gleicher Weise wie native im Monitorsystem und in den Gastsystemen unter BS2000/OSD ab V7.0 eingesetzt werden. Der VM2000-Administrator muss bei der Geräte- und Pubset-Zuordnung zu den beteiligten VMs und Gastsystemen Vorkehrungen für den Einsatz von SPACEPRO treffen. Für nähere Informationen dazu siehe Handbuch „VM2000“ [62].

#### Verwaltung der Metadaten

Für den Ablauf benötigt bzw. verwaltet SPACEPRO verschiedene Arten von Metadaten, die im Folgenden ausführlich beschrieben werden:

a) SPACEPRO-Parameter (systemlokal)

b) SPACEPRO-Optionen (pro Pubset)

c) SPACEPRO-Historie (pro Pubset)

d) Jobvariablen zur Verwaltung von SPACEPRO-Sperren (pro Pubset)

e) Diagnosedaten (systemlokal)

*zu a) SPACEPRO-Parameter*

Die SPACEPRO-Parameter enthalten systemglobale Einstellungen für den Ablauf von SPACEPRO. Sie steuern im Einzelnen (in Klammern jeweils die Einstellung nach Installation bzw. nach einem Reset von SPACEPRO angegeben):

- Verhalten im Lock-Konfliktfall beim Zugriff auf SPACEPRO-Optionen und in Erweiterungs- und Reduktionsaktionen auf den einzelnen SPACEPRO-Pubsets
  - LOCK-WAIT-TIME: Wartezeit bis zur Wiederholung einer Aktion (Standard: 60 s)
  - LOCK-RETRIES: Maximale Anzahl von Wiederholungen (Standard: 3)
- Standardwerte für gleichnamige Operanden in den Kommandos MODIFY-PUBSET-SPACEPRO-OPTIONS bzw. MOVE-SPACEPRO-DISK
  - Vorgabe des Operanden SATURATION-LEVEL (Standard: 3)
  - Vorgabe des Operanden BLOCKING-TIME (Standard: 24 Stunden)
  - Vorgabe des Operanden CHECK-PUBSET-MIRRORS (Standard: *NO)
  - Vorgabe des Operanden CLEAR-VOLUME-TIME (Standard: 300 Sekunden)
- Einstellungen für die PROP-XT-Schnittstelle von SPACEPRO:
  - OPERATOR-USER-ID: Benutzerkennung für den Ablauf von PROP-XT-Aktionen (Standard: SYSOPR)
  - OPERATOR-PASSWORD: Kennwort der Benutzerkennung (Standard: *NO)
  - OPERATOR-ROLE: Operator-Rolle für die PROP-XT-Schnittstelle (Standard: SYSADM)

Die SPACEPRO-Parameter gelten nur systemlokal. Dies ist insbesondere zu beachten, wenn ein Shared-Pubset als SPACEPRO-Pubset eingerichtet wird.

Die SPACEPRO-Parameter werden auf dem Home-Pubset in der PLAM-Bibliothek `$TSOS.SYSPAR.SPACEPRO.010` abgelegt. Die Bibliothek wird implizit beim ersten Zugriff z.B. mit dem Kommando SHOW- bzw. MODIFY-SPACEPRO-PARAMETERS angelegt.

*zu b) SPACEPRO-Optionen*

Mit dem Kommando SHOW-PUBSET-SPACEPRO-OPTIONS werden die Optionen angezeigt.

Die SPACEPRO-Optionen enthalten für den jeweiligen Pubset Einstellungen für seine Überwachung und für die Durchführung von SPACEPRO-Aktionen. Die eingestellten Optionen wertet der SPACEPRO-Monitor während der Pubset-Überwachung aus. Mit Ausnahme der Sättigungsstufe werden die Einstellungen auch bei der Überwachung mit dem INSPECTOR berücksichtigt. Folgende Optionen können mit dem Kommando MODIFY-PUBSET-SPACEPRO-OPTIONS festgelegt bzw. geändert werden:

- Zuordnung eines Pool-Pubsets (Operand POOL-PUBSET)
- Sättigungsstufe als Auslösekriterium für die automatische Erweiterung des Pubsets (Operand SATURATION-LEVEL)

- Mindestzeit zwischen zwei automatischen Erweiterungen als Voraussetzung für die Durchführung der nächsten automatischen Erweiterung (Operand BLOCKING-TIME)
- Angabe zur Homogenitätsprüfung bezüglich Pubset-Spiegelung bei Durchführung einer Pubset-Erweiterung (Operand CHECK-PUBSET-MIRRORS)
- Beschreibungstext zu den SPACEPRO-Optionen (Operand REMARK)

Standardmäßig sind folgende SPACEPRO-Optionen voreingestellt:

- Es ist kein Pool-Pubset zugeordnet (POOL-PUBSET=*NONE). Damit ist die automatische Erweiterung zunächst ausgeschaltet.
- Die Einstellungen für die Sättigungsstufe, die Mindestzeit zwischen zwei automatischen Erweiterungen und die Homogenitätsprüfung bei Pubset-Erweiterung sollen jeweils den SPACEPRO-Parametern entnommen werden (entspricht dem Operandenwert *BY-PARAMETER).
- Ein Beschreibungstext ist noch nicht vorhanden (REMARK=*NONE).

Die SPACEPRO-Optionen eines Pubets sind auf dem jeweiligen Pubset in folgender Datei abgelegt:
`:<catid>:$TSOS.SYSDAT.SPACEPRO.OPTION`

Diese Datei wird implizit erstellt mit dem ersten SHOW- bzw. MODIFY-PUBSET-SPACEPRO-OPTIONS-Kommando für den betroffenen Pubset.

*zu c) SPACEPRO-Historie*

Die SPACEPRO-Historie protokolliert die von SPACEPRO automatisch durchgeführten Pubset-Erweiterungen/-Reduktionen, die mit Fehler abgebrochenen Erweiterungen/Reduktionen sowie die manuell über MOVE-SPACEPRO-DISK-Kommandos durchgeführten Erweiterungen/Reduktionen. Angezeigt werden SPACEPRO-Pubset (bzw. Volume-Set), Zeitpunkt der Erweiterung bzw. Reduktion, zugeordneter Pool-Pubset und Volume (alte VSN und MN). SPACEPRO analysiert diese Daten zur Überprüfung der SPACEPRO-Option BLOCKING-TIME.

Die Systembetreuung kann sich mit dem Kommando SHOW-PUBSET-SPACEPRO-HISTORY über die protokollierten Aktionen informieren.

Die SPACEPRO-Historie ist auf dem jeweiligen Pubset in folgender Datei abgelegt:
`:<catid>:$TSOS.SYSDAT.SPACEPRO.HISTORY`

Diese Datei wird implizit erstellt mit der ersten automatisch durchgeführten Erweiterung bzw. Reduktion des betroffenen Pubsets oder mit der ersten manuellen Erweiterung bzw. Reduktion mit dem Kommando MOVE-SPACEPRO-DISK.

*zu d) Jobvariablen zur Verwaltung von SPACEPRO-Sperren*

Zur Verwaltung von Sperren für die SPACEPRO-Optionen eines Pubsets und die Aktionen „Erweiterung“ bzw. „Reduktion“ eines Pubsets legt SPACEPRO auf dem betroffenen Pubset folgende Jobvariablen an:

- `:<catid>:$TSOS.SYSLCK.SPACEPRO.OPTION`
- `:<catid>:$TSOS.SYSLCK.SPACEPRO.HOST`

*zu e) Diagnosedaten*

Zu Diagnosezwecken legt SPACEPRO auf dem Home-Pubset weitere Dateien und Jobvariablen an:

- Loggingdateien mit dem Pfadnamen `$TSOS.SYSTRC.SPACEPRO.*`
- Monitor-Jobvariablen mit dem Pfadnamen `$TSOS.MONJV.SPACEPRO.*`



*Sicherung und Bearbeitung der Metadaten*

Die Dateien für SPACEPRO-Optionen, -Historie und -Parameter sind in die Datensicherung des jeweiligen Pubsets einzubeziehen. Die Jobvariablen zur Lock-Verwaltung sowie die zu Diagnosezwecken angelegten Loggingdateien und Monitor-Jobvariablen müssen nicht gesichert werden.



**ACHTUNG!**
Die in a) bis d) genannten Dateien und Jobvariablen können ausschließlich von SPACEPRO bearbeitet werden. Manuelles Editieren kann sie zerstören.

Bei grundsätzlichen Problemen können diese Dateien und/oder Jobvariablen entweder insgesamt oder auf einem bestimmten Pubset gelöscht werden. Dies entspricht einem Zurücksetzen von SPACEPRO (bzgl. der Parameter auf den Installationszeitpunkt, bzgl. der Optionen und Historie auf den Zeitpunkt der Pubset-Erstellung).

### 9.5.6.1 Konfiguration für SPACEPRO

Da das Subsystem SPACEPRO Funktionen der Software-Produkte PROP-XT und JV nutzt, ist es nur zusammen mit diesen Software-Produkten ablauffähig.

Pubsets, die mit DRV gespiegelt werden und Pubsets, die aus GS-Volumes bestehen, werden nicht von SPACEPRO unterstützt.

Der SPACEPRO-Monitor setzt nur die Standardprivilegien der Benutzerkennung TSOS voraus. Das gilt auch für die Ablaufumgebung der Host-Prozeduren, wenn statt des Monitors die Pubset-Überwachung mit openSM2 genutzt wird (siehe Abschnitt „Pubset-Überwachung mit INSPECTOR“ auf Seite 390).

SPACEPRO nutzt grundsätzlich Kommandos der Geräte- und Pubset-Verwaltung mit dem Privileg STANDARD-PROCESSING, PROP-ADMINISTRATION und TSOS. Bei Bedarf (und optional) nutzt es Kommandos der Subsystem- und Benutzerverwaltung, soweit diese der Benutzerkennung TSOS zugewiesen sind.

*PROP-XT-Schnittstelle*

Für das Zu- und Wegschalten von Platten nutzt SPACEPRO die PROP-XT-Schnittstelle, die über entsprechende Berechtigungen verfügen muss. Dazu benötigt die PROP-XT-Schnittstelle eine Benutzerkennung mit Operatorrolle, das zugehörige LOGON-Kennwort und die zugehörige Operatorrolle. Diese Daten werden als SPACEPRO-Parameter verwaltet. Im Auslieferungszustand bzw. nach einem Reset sind folgende Werte eingestellt:

- Benutzerkennung SYSOPR
- LOGON-Kennwort *NO
- Operatorrolle SYSADM

Die Benutzerkennung darf nicht gesperrt sein (ggf. ist das Kommando UNLOCK-USER unter der Benutzerkennung SYSPRIV erforderlich). Der Zugang muss mit dem angegebenen LOGON-Kennwort möglich sein und sie muss über die angegebene Operatorrolle verfügen.

Die Operatorrolle muss die Berechtigung für die Kommandos ATTACH-DEVICE und DETACH-DEVICE (Privileg OPERATING), für PROP-XT-Kommandos und für den Empfang der von SPACEPRO überwachten Meldungen besitzen. Folgende Meldungen überwacht SPACEPRO über die PROP-XT-Schnittstelle:
`DMS1400, EXC0440 ... EXC0445, DMS037F, DMS038F, DMS038E, DMS038C, DMS0381, NKVD013`

Es muss sichergestellt werden, dass diese Konsolmeldungen der PROP-XT-Instanz von SPACEPRO zugestellt werden. Insbesondere dürfen diese Konsolmeldungen nicht unterdrückt werden.

*Pubset reduzieren*

Bei der Reduktion eines Pubsets, die entweder über das Kommando MOVE-SPACEPRO-DISK mit FROM=*PUBSET(...) oder durch openSM2 ausgelöst wird, muss SPACEOPT (siehe Seite 393) das zu entnehmende Volume entleeren, falls es Belegungen aufweist. Das Entleeren erfolgt über einen internen Aufruf des Subsystems SPACEOPT. Wenn SPACEOPT nicht verfügbar ist, wird die Reduktion mit Fehler beendet.
SPACEOPT nutzt auf dem betroffenen Pubset die Benutzerkennung SYSSOPT.

### Hinweise zur Pubset-Konfiguration

Aus Verfügbarkeitsgründen sollten Pubsets (einschließlich Pool-Pubsets) in einem Speicherplattensystem nicht box-übergreifend konfiguriert werden. Im Allgemeinen ist ein Pool-Pubset pro Box ausreichend, vor allem dann, wenn alle Pubsets in der Box aus Volumes mit identischen Betriebsparametern bestehen.

In speziellen Fällen kann es notwendig oder sinnvoll sein, mehrere Pool-Pubsets in einer Box bereitzustellen.
Beispiel: Es werden Volumes mit unterschiedlichen Betriebsparametern in einer Box genutzt und diese können nicht im BS2000-System kontrolliert werden.

Die autonome Erweiterung eines Pubsets mit SPACEPRO schlägt fehl, wenn die Betriebsparameter des Pubsets aus BS2000-Sicht unverträglich mit sind zu denen des Pool-Pubsets.

*Homogenität bezüglich Pubset-Spiegelung überprüfen*

Eine Homogenitätsprüfung bezüglich Pubset-Spiegelung (Angabe des Operanden CHECK-PUBSET-MIRRORS=*YES des Kommandos MOVE-SPACEPRO-DISK bzw. Einstellung der entsprechenden SPACEPRO-Option) ist nur sinnvoll, wenn die folgenden Bedingungen erfüllt sind:

- Alle Pubset-Platten liegen im gleichen Plattenspeichersystem
- Das Subsystem SHC-OSD ist verfügbar

SPACEPRO prüft diese Bedingungen nicht ab. Beim internen Aufruf des Kommandos MODIFY-PUBSET-PROCESSING wird diese Operandenangabe ignoriert, wenn SHC-OSD nicht verfügbar ist.

Die Homogenitätsprüfung der Pubset-Spiegelung beinhaltet auch einen Box-Check.

Zusätzliche Hinweise zur Erweiterung von gespiegelten SPACEPRO-Pubsets:

1. Eine Pubset-Erweiterung mit Homogenitätsprüfung kann nur dann erfolgreich ausgeführt werden, wenn der zugeordnete Pool-Pubset zum Zeitpunkt der Erweiterung ein Volume mit den aktuellen Spiegeleigenschaften des zu erweiternden Pubsets enthält und dieses homogen ist (Kriterien siehe Seite 370).

2. Wenn der aktive BCV-Spiegel eines Pubset genutzt werden soll (z.B. für eine HSMS-Sicherung von den Spiegelplatten mit der Anweisung BACKUP-FILES), müssen auch die Spiegelplatten zugeschaltet sein. Das Zuschalten geschieht immer kommandogesteuert, wobei  Pubset-Erweiterungen berücksichtigt werden müssen. Das gleiche gilt ggf. analog für die VM2000-Zuweisungen der Spiegel an ein Gastsystem, das sie nutzen will.
3. Die Re-Synchronisierung von Pubset-Spiegeln ist nur möglich, wenn der Pubset zwischenzeitlich nicht erweitert wurde. Dies gilt sowohl für die automatische Re-Synchronisierung des Pubset-Spiegels nach einer HSMS-Sicherung (Operand RESUME-MIRRORING=*YES in der BACKUP-FILES-Anweisung) als auch für die manuell veranlasste Re-Synchronisierung des abgetrennten Pubset-Spiegels mit dem Kommando RESUME-MULTI-MIRRORING.

In SPACEPRO kann zur Absicherung der Re-Synchronisierung (siehe Hinweis 3) die automatische Erweiterung eines Pubsets vorübergehend deaktiviert werden. Dazu wird die SPACEPRO-Option POOL-PUBSET dieses Pubset vorübergehend auf *NONE gesetzt und nach erfolgter Synchronisation wird die automatische Erweiterung wieder eingeschaltet durch Zuweisen des ursprünglichen Pool-Pubsets. Bei Bedarf kann nach einer Re-Synchronisierung der Pubset auch manuell erweitert werden mit Hilfe des Kommandos MOVE-SPACEPRO-DISK.

Spiegelplatten von SPACEPRO- und Pool-Pubsets sollten im „Normalbetrieb“ nicht zugeschaltet (im Zustand DETACHED) sein, damit Pubset-Import-Vorgänge nicht auf Konsolfragen laufen. Das Zuschalten ist nur zum Zwecke einer (vorübergehenden) Nutzung vorzunehmen.

### SPACEPRO für Shared-Pubsets

SPACEPRO unterstützt Shared-Pubsets, wenn alle beteiligten Systeme BS2000/OSD ab V7.0 und MSCF ab V16.0 einsetzen. Bei einer Pubset-Erweiterung (automatisch oder mit dem Kommando MOVE-SPACEPRO-DISK) wird das neu hinzukommende Volume implizit an allen Pubset-Sharern zugewiesen. Für Pubset-Sharer, die nach einer Erweiterung hinzukommen, ist die „Regel 3“ auf Seite 387 zu beachten. Bei einer Pubset-Reduktion (mit dem Kommando MOVE-SPACEPRO-DISK) wird das entnommene Volume implizit an allen Pubset-Sharern weggeschaltet.

### Pubset-Kopien

Ein Pubset, der mit dem Dienstprogramm PVSREN (mit TimeFinder-, Snapview- oder DRV-Funktionen) erzeugt wird, besitzt die gleichen SPACEPRO-Optionen und die gleiche SPACEPRO-Historie wie der Original-Pubset. Je nach Konfiguration kann es notwendig sein, die SPACEPRO-Optionen zu ändern. Die SPACEPRO-Historie lässt sich durch Löschen der History-Datei zurücksetzen.

### 9.5.6.2 Regeln zum Generieren von Volumes und Zuschalten von Pubsets

Wenn SPACEPRO im Rechnerverbund eingesetzt wird, müssen alle Volumes der SPACEPRO-Pubsets und der gemeinsam genutzen Pool-Pubsets auf allen beteiligten BS2000-Systemen den gleichen mnemotechnischen Gerätename besitzen (dies wird auch unabhängig von SPACEPRO für alle Geräte empfohlen).

Da in den SPACEPRO-Optionen die Pool-Pubsets nur mit ihrer Katalogkennung eingetragen werden und die MNs der einzelnen Volumes somit nicht bekannt sind, werden die Volumes der Pool-Pubsets über das Kommando ATTACH-DEVICES mit Angabe der Katalogkennung zugeschaltet:

```
/ATTACH-DEVICES UNIT=*PUBSET-DEVICES(PUBSET=<catid>)
```

Die zugehörigen MNs der Pubset-Volumes werden in diesem Fall aus dem SVL der Pubres bzw. der Volres des Control-Volume-Sets ermittelt. Dazu muss der Pool-Pubset mit seiner Pubres/Volres im MRSCAT eingetragen sein.

Damit eine autonome Pubset-Erweiterung beim nächsten BS2000-Systemstart korrekt erkannt und berücksichtig wird, müssen (soweit erforderlich) Volumes von SPACEPRO-Pubsets ebenfalls über das Kommando ATTACH-DEVICES mit Angabe der Katalogkennung zugeschaltet werden. Alternativ kann hier auch das Kommando ATTACH-DEVICES mit Angabe der Pubres-Mnemonik verwendet werden:

```
/ATTACH-DEVICES UNIT=*PUBSET-DEVICES(PUBSET=*BY-PUBRES-DEVICE(UNIT=<mn>))
```

Für das Generieren der Volumes sind folgende zwei Regeln zu beachten:

**Regel 1** Alle Volumes von SPACEPRO- und Pool-Pubsets müssen auf den beteiligten Systemen die gleichen MNs besitzen.

**Regel 2** Alle Pool-Pubsets müssen auf allen beteiligten System mit dem Kommando ATTACH-DEVICE und Angabe der Katalogkennung zuschaltbar sein.
Alle SPACEPRO-Pubsets müssen auf allen beteiligten System mit dem Kommando ATTACH-DEVICE und Angabe der Katalogkennung bzw. Pubres-Mnemonik zuschaltbar sein.

Wenn im Rechnerverbund die autonome Erweiterung eines Pubsets an anderen Systemen nicht beachtet wird, können folgende Probleme auftreten:

- Der Import eines erweiterten Pubsets in einem zweiten System schlägt fehl, wenn die Volumes bereits vor der Erweiterung zugewiesen waren, aber das Erweiterungs-Volume nicht zugewiesen ist.
- Ein erweiterter Pubset wird bei der Systemeinleitung vor seinem Import referenziert (z.B. Angabe einer Rep- oder Parameter-Datei). Home-Pubset und Paging-Pubsets dürfen referenziert werden, wobei bei einem Paging-Pubset der IPL ggf. einen Online-Scan nachholt.

Zur Vermeidung dieser Probleme sollten noch folgende zwei Regeln beachtet werden:

**Regel 3** Alle SPACEPRO-Pubsets sollten (soweit notwendig) unmittelbar vor dem Importieren nach Regel 2 zugeschaltet werden.

**Regel 4** SPACEPRO-Pubsets dürfen in der Systemeinleitung vor ihrem Import nicht referenziert werden, außer als Home- oder Paging-Pubset. Nach Erweiterung eines Paging-Pubsets muss in der nachfolgenden Systemeinleitung mit einem Online-Scan gerechnet werden.

### 9.5.6.3 Einrichten von Pubsets für SPACEPRO

In der Praxis besteht die Aufgabe darin, SPACEPRO in einer bereits bestehenden Platten- und Pubset-Landschaft zu konfigurieren. Dies geschieht in folgenden Schritten:

1. Platten, die für den Free-Pool in Frage kommen, auswählen. Dabei sollten Regel 1, Regel 2 und die „Hinweise zur Pubset-Konfiguration“ auf Seite 384 beachtet werden.
2. Pubset-Struktur für den Free-Pool festlegen. Ziel sollte sein, mit möglichst wenig Pool-Pubsets auszukommen (z.B. ein Pool-Pubset pro Box).
3. Pool-Pubsets mit SIR erstellen. Es ist wichtig, den Pool-Pubset nur als „Gerüst“ zu erstellen. Das bedeutet, dass ein SM-Pubset nur einen Control-Volume-Set mit einer Volres und ein SF-Pubset nur eine Pubres enthält. Weitere Volumes bzw. Volume-Sets dürfen nicht hinzugefügt werden, da SIR keine Allokierungssperre setzen kann. Die Allokierungssperre ist aber notwendig für die problemlose Entnahme eines Volumes.
4. Pool-Pubset mit freien Volumes auffüllen. Die Volumes werden mit dem Kommando MOVE-SPACEPRO-DISK und Allokierungssperre (Operand ALLOCATION-ON-VOLUME=*NOT-ALLOWED) dem Pool-Pubset hinzugefügt. Die Volumes werden dabei formatiert und können optional vollständig geleert werden:
```
/MOVE-SPACEPRO-DISK FROM=*DISK(UNIT=<mn>, OVERWRITE-VSN=<vsn>),
      TO=*PUBSET(PUBSET=<pool-pubset>,ALLOCATION-ON-VOLUME=*NOT-ALLOWED)
      [,DESTROY-OLD-CONTENTS=YES]
```
5. Danach kann der Pool-Pubset exportiert und weggeschaltet werden.
6. Pool-Pubset im MRSCAT eintragen. An allen Systemen, die den Pool-Pubset nutzen sollen, muss dieser im MRSCAT eingetragen werden. Er muss mit dem Kommando ATTACH-DEVICES unter Angabe der Katalogkennung zuschaltbar sein. Dies sollte durch Testen sichergestellt werden (einmal zuschalten und importieren).
7. Die SPACEPRO-Parameter den Kundenanforderungen entsprechend an jedem beteiligten System einstellen. Dabei ist besonders auf Parameter zu achten, die bei den SPACEPRO-Optionen über den Operandenwert *BY-PARAMETER referenziert werden.
8. In allen Kommando- und Konfigurationsprozeduren (z.B. CMDFILE) die Zu- und Wegschaltungen von Geräten der SPACEPRO-Pubsets überprüfen (und ggf. umstellen). Bei VM2000-Betrieb gilt dies entsprechend für die VM2000-Konfigurationsprozeduren (für VM2000 sind Details im Handbuch „VM2000“ [62] beschrieben).
9. Bei allen Pubsets, die autonom erweitert werden sollen, muss ein Pool-Pubset in den SPACEPRO-Optionen zugewiesen werden.

10. Die SPACEPRO-Konfiguration unbedingt vorher testen:

    – Während der Überwachung mit dem SPACEPRO-Monitor können mit dem Kommando SIMULATE-SPACEPRO-EVENT reale Pubset-Erweiterungen angestoßen werden. Diese lassen sich mit dem Kommando MOVE-SPACEPRO-DISK leicht rückgängig machen, da die Erweiterung mit Allokierungssperre erfolgt.

    – Für die Pubset-Erweiterungen, die durch openSM2-Events ausgelöst werden sollen, kann ebenfalls eine Erweiterung zu Testzwecken eingestellt werden (siehe Abschnitt „Pubset-Überwachung mit INSPECTOR“ auf Seite 390).

#### 9.5.6.4 Pubset-Überwachung mit SPACEPRO-Monitor

Die Pubset-Überwachung durch den SPACEPRO-Monitor wird mit dem Kommando START-SPACEPRO-MONITORING aktiviert und kann mit dem Kommando STOP-SPACEPRO-MONITORING beendet werden. Das Kommando SHOW-SPACEPRO-STATUS gibt Auskunft über den aktuellen Zustand des SPACEPRO-Monitors.

- Ausgabe bei nicht gestartetem Monitor:

```
SPACEPRO <version> (<date>)
SPACEPRO-MONITOR-STATUS:   NOT STARTED
```

- Ausgabe bei gestartetem Monitor:

```
SPACEPRO <version> (<date>)
SPACEPRO-MONITOR-STATUS:   STARTED
```

Das Kommando START-SPACEPRO-MONITORING wird auf Konsole protokolliert:

```
<R %@001-000.135657  %  SPP0001 PROCEDURE 'START-SPACEPRO-MONITOR', VERSION
'<version>' OF '<date>' STARTED FROM FILE ':4V25:$TSOS.SYSLNK.SPACEPRO.ver'
<R %@001-000.135658  %  SPP0318 MONITORING STARTED FOR SPACEPRO EVENTS
```

Wenn ein eingestellter Sättigungsschwellwertes eines SPACEPRO-Pubsets erreicht ist, werden die SPACEPRO-Aktionen auf Konsole protokolliert:

```
<* %0FV5-000.141711  %  EXC0443 SPACE SATURATION LEVEL '3' REACHED ON PVS
'KMSV'. SPACE USED = 1202157 HP'S, AVAILABLE SPACE = 12402 HP'S
...
  <R %@001-000.141726  %  SPP0425 FORMER POOL VOLUME 'POLV.2' FORMATTED AS
VOLUME 'KMSV.1'
  <R %@001-000.141727  %  SPP0327 EXPANSION OF PUBSET 'KMSV' THROUGH POOL-
PUBSET 'POLS' COMPLETED
<* %0FV6-000.141727  %  EXC0440 SPACE SATURATION LEVEL '0' REACHED ON PVS
'KMSV'. SPACE USED = 1202163 HP'S, AVAILABLE SPACE = 1226955 HP'S
```

Nach Durchführung der SPACEPRO-Aktionen für den Pubset KMSV liefert das Kommando SHOW-PUBSET-SPACEPRO-HISTORY folgende Informationen:

```
/SHOW-PUBSET-SPACEPRO-HISTORY PUBSET=KMSV
===========================================================================
FCT T-PS T-VS/N       TIME(UTC)      F-PS F-VS/N MNEM MSG
===========================================================================
                  HISTORY OF PUBSET KMSV
EXP KMSV KMSV.1 <date> <time>        POLS POLV.2 E08C SPP0327
```

### 9.5.6.5 Pubset-Überwachung mit INSPECTOR

Die Software-Komponente INSPECTOR des Produkts openSM2 bietet innerhalb der Überwachungsfunktionen für BS2000/OSD-Systeme auch Funktionen zur Pubset-Überwachung: Bei Über- oder Unterschreiten eines festgelegten Wertes einer Messgröße kann eine Host-Prozedur in dem betroffenen BS2000-System gestartet werden.
Für das Auslösen von SPACEPRO-Aktionen über eine Host-Prozedur muss der Pubset-Füllungsgrad (gemessen als prozentuale Pubset-Belegung oder durch Sättigungsstufen) überwacht werden. Dabei kann das Überschreiten eines bestimmten Füllungsgrades eine Pubset-Erweiterung, das Unterschreiten eine Pubset-Reduzierung auslösen.

#### Vorbereitungen

Im BS2000-System muss im Subsystem openSM2 die Pubset-Überwachung eingeschaltet werden. Die openSM2-Tools sind zu aktivieren und für die Komponente INSPECTOR sind die beidem Kommunikationspartner INSPECTOR-Agent (in BS2000/OSD) und INSPECTOR-Manager (im PC) zu starten.

Im INSPECTOR-Manager definiert die Systembetreuung Regeln für die zu überwachenden Pubsets. Sie legt z.B. in einer Regel einen Pubset-Füllungsgrad fest, bei dessen Überschreitung eine Host-Prozedur zu starten ist, die eine Pubset-Erweiterung mit SPACEPRO auslöst. Genauso kann sie auch eine Regel definieren, die bei Unterschreiten eines Füllungsgrades eine Host-Prozedur zur Pubset-Reduzierung startet.

Details zur Konfiguration von openSM2 und der Komponente INSPECTOR sowie der Regelbildung sind im Handbuch „openSM2“ [51] beschrieben.

Wenn der INSPECTOR die Pubsets-Überwachung übernimmt, sollte der SPACEPRO-Monitor nicht gestartet werden.

### Host-Prozeduren für SPACEPRO-Aktionen

In der Bibliothek `SYSLNK.SPACEPRO.010` stehen folgende zwei vorgefertigten Host-Prozeduren zur Verfügung:

- `HOSTPROC-SPP-EXPAND` soll beim Überschreiten eines festgelegten Pubset-Füllungsgrades eine Pubset-Erweiterung über SPACEPRO auslösen. Optional kann die Erweiterung auch zu Testzwecken ausgelöst werden.
- `HOSTPROC-SPP-REDUCE` soll beim Unterschreiten eines festgelegten Pubset-Füllungsgrades eine Pubset-Reduzierung über SPACEPRO auslösen.

Die ausgelieferten Host-Prozeduren müssen ggf. den Kundenbedürfnissen angepasst werden. Insbesondere ist zu beachten, dass eine Pubset-Reduzierung nicht in einer Hochlast-Situation ausgelöst werden sollte. Da openSM2 ggf.auch die I/O-Last überwacht, kann dieser Fall über zusätzliche Regeln im INSPECTOR abgefangen werden.

Beide Host-Prozeduren melden über den Aufruf der Prozedur `REPORT-SM2-EVENT` das SM2-Event „Überschreiten“ bzw. „Unterschreiten eines Pubset-Füllungsgrades“ an SPACEPRO. Die Schnittstelle von Host-Prozeduren zu SPACEPRO besteht aus folgendem Prozeduraufruf:

```
/CALL-PROCEDURE FROM-FILE=*LIBRARY-ELEMENT(LIBRARY=$TSOS.SYSLNK.SPACEPRO.010,
    ELEMENT=REPORT-SM2-EVENT,TYPE=*STD),PROCEDURE-PARAMETERS=(...)
```

Dabei werden über die Prozedurparameter werden folgende Informationen an SPACEPRO übergeben:

| **Prozedurparameter** | **mögliche Werte** | **Bedeutung** |
|---|---|---|
| MEASURE= | PUBALLOC | Auswertung der Messgröße `PUBALLOC` (Pubset-Belegung in %) |
| | PUBSATL | Auswertung der Messgröße `PUBSATL` (erreichte Sättigungsstufe des Pubsets) |
| EVENT= | EXCEED | „Überschreiten des Wertes“ erfordert Pubset-Erweiterung (`HOSTPROC-SPP-EXPAND`) |
| | FALL-BELOW | „Unterschreiten des Wertes“ erfordert Pubset-Reduzierung (`HOSTPROC-SPP-REDUCE`) |
| | TEST-EXCEED | wie `EXCEED`, aber die Pubset-Erweiterung soll für Testzwecke erfolgen (ist als auskommentierte Option in `HOSTPROC-SPP-EXPAND` enthalten) |
| PUBSET= | <pubset-id> | Katalogkennung des überwachten Pubsets |
| VOLSET= | *NONE<br><volset-id> | bei Pubset-Typ SF<br>Id des Control-Volume-Sets bei SM-Pubsets |
| SATURATION= | <integer 1..99> | Prozentsatz der Pubset-Belegung bzw. Sättigungsstufe (abhängig von der ausgewerteten Messgröße) |

| Prozedurparameter | mögliche Werte | Bedeutung |
|---|---|---|
| RC= | <s-variable> | Name der S-Variable, in der der Returncode zurückgeliefert werden soll: im Fehlerfall ein Wert ungleich 0 (in beiden Host-Prozeduren die S-Variable RC) |

Bei Auftreten eines SM2-Events (siehe Parameter EVENT) gibt SPACEPRO folgende Meldung auf Konsole aus:

```
SPP0343 SM2 EVENT '(&00)' OF MEASURE '(&01)' REPORTED
```

Weitere Meldungen gibt SPACEPRO abhängig von der ausgewerteten Messgröße aus.

Bei Auswertung der prozentualen Pubset-Belegung (Messgröße `PUBALLOC`):

```
SPP0345 PUBSET ALLOCATION OF ''(&&00)'' PER CENT REACHED ON VOLSET '(&01)' OF
SM-PUBSET '(&02)'
SPP0346 PUBSET ALLOCATION OF ''(&&00)'' PER CENT REACHED ON SF-PUBSET '(&01)'
SPP0347 'PUBSET ALLOCATION OF VOLSET ''(&&01)'' IN SM-PUBSET ''(&&02)''
FALLED DOWN TO ''(&&00)'' PER CENT' ,
SPP0348 PUBSET ALLOCATION OF SF-PUBSET '(&01)' FALLED DOWN TO ''(&&00)'' PER
CENT
```

Bei Auswertung der Sättigungsstufe (Messgröße `PUBSATL`):

```
SPP0324 SATURATION LEVEL '(&00)' REACHED ON VOLSET '(&01)' OF SM-PUBSET
'(&02)'
SPP0325 SATURATION LEVEL '(&00)' REACHED ON SF-PUBSET '(&01)'
```

Anschließend führt SPACEPRO in Abhängigkeit vom gemeldeten SM2-Event folgende Aktionen durch:

- Bei `EVENT=EXCEED` untersucht SPACEPRO die SPACEPRO-Optionen und -Historie für den gemeldeten Pubset und führt ggf. eine Pubset-Erweiterung durch. Die in den SPACEPRO-Optionen eingestellte Sättigungsstufe wird nicht berücksichtigt (gilt nur für die Überwachung durch den SPACEPRO-Monitor).
- Bei `EVENT=FALL-BELOW` reduziert SPACEPRO den gemeldeten Pubset bzw. Volume-Set um ein Volume und integriert dieses in den zugehörigen Pool-Pubset.
  Die Reduktion wird unabhängig von vorangegangenen Reduktionen bzw. Erweiterungen des Pubset durchgeführt. Die Reduktion schlägt fehl, wenn das Volume nicht vollständig geleert werden kann (interner Aufruf der SPACEOPT-Funktion CLEAR-VOLUME, siehe Handbuch „SPACEOPT“ [54]).
- Bei `EVENT=TEST-EXCEED` führt SPACEPRO wie bei `EVENT=EXCEED` eine reale Pubset-Erweiterung durch. Das Hinzufügen eines Volumes erfolgt jedoch mit Allokierungssperre (`ALLOCATION-ON-VOLUME=*NOT-ALLOWED`). Die Erweiterung lässt sich deshalb mit dem Kommando MOVE-SPACEPRO-DISK einfach wieder rückgängig machen. Das SM2-Event dient zum Testen einer Pubset-Erweiterung über openSM2. Es wird bei entsprechender Anpassung der Server-Prozedur `HOSTPROC-SPP-EXPAND` statt `EXCEED` gemeldet.

Die weiteren Aktionen und Konsolmeldungen entsprechen bei einer Pubset-Erweiterung denen des SPACEPRO-Monitors (siehe Seite 389), bei einer Pubset-Reduktion erscheint als Erfolgsmeldung auf Konsole:

```
<R %@001-000.141308  %  SPP0334 REDUCTION OF PUBSET 'KMSV' THROUGH POOL-
PUBSET 'POLS' COMPLETED
```

Danach liefert das Kommando SHOW-PUBSET-SPACEPRO-HISTORY für den Pubset KMSV folgende Informationen:

```
/SHOW-PUBSET-SPACEPRO-HISTORY PUBSET=KMSV
===========================================================================
FCT T-PS T-VS/N        TIME(UTC)      F-PS F-VS/N MNEM MSG
===========================================================================
                  HISTORY OF PUBSET KMSV
EXP KMSV KMSV.1  <date> <time>        POLS POLV.2 E08C SPP0327
RED KMSV KMSV.1  <date> <time>        POLS POLV   E08C SPP0334
```

## 9.5.7 Überwachung der Speicherplatz-Sättigung

Um Engpässe an Speicherplatz auf gemeinschaftlichen Datenträgern rechtzeitig erkennen zu können, überwacht das System den Sättigungsgrad von Pubsets. Die Systembetreuung beeinflusst diese Überwachung durch die Vereinbarung von Sättigungsstufen. Diese werden für SF-Pubsets pubset-spezifisch festgelegt, während bei SM-Pubsets die Festlegungen volume-set-spezifisch sind. Ein Sättigungsgrad ist erreicht, wenn auf einem SF-Pubset bzw. Volume-Set weniger Platz frei ist, als für die entsprechende Sättigungsstufe festgelegt worden ist.

Bei der Änderung des erreichten Sättigungsgrads werden die im Folgenden beschriebenen Meldungen an der Bedienstation ausgegeben. Außerdem haben die verschiedenen Sättigungsgrade Einfluss auf die Behandlung von Speicherplatzanforderungen.

### Sättigungszustand

Meldung bei SF-Pubsets:

```
EXC044i    SPACE SATURATION LEVEL 'i' REACHED ON PVS '(&00)'.
           SPACE USED = (&01) HP'S, AVAILABLE SPACE = (&02) HP'S.
```

`i` ist eine Ziffer von 0 bis 5 und bezeichnet den Grad der Sättigung.

Meldung beim Überschreiten des Zip-Levels auf einem SF-Pubset:

```
DMS141C    SPACE SATURATION LEVEL 'ZIP' REACHED ON PUBSET '(&00)'.
           SPACE USED = (&01) HP'S, AVAILABLE SPACE = (&02) HP'S.
           IMMEDIATELY ERASE FILES NO LONGER REQUIRED
```

Meldung für Volume-Sets eines SM-Pubsets:

```
DMS1400    SPACE SATURATION LEVEL '(&03)' REACHED ON VOLUMESET '(&00)'
           OF PUBSET '(&04)'. SPACE USED = (&01) HP'S,
           AVAILABLE SPACE = (&02) HP'S
```

`(&03)` ist eine Ziffer von 0 bis 6 und bezeichnet den Grad der Sättigung, wobei 6 für den Zip-Level steht.

| Grad | Auswirkungen im System | Maßnahmen des Operators |
|---|---|---|
| 0 | Normaler Systemablauf | Keine |
| 1-3 | Vorwarnung wegen beginnender Sättigung für einen Pubset bzw. Volume-Set | Systembetreuung verständigen; die Dialogteilnehmer mit dem Kommando INFORM-ALL-JOBS zum Löschen von nicht benötigten Dateien auffordern |
| 4 | Speicherplatzanforderungen von Benutzer-Jobs für einen Pubset bzw. Volume-Set werden mit Meldung `DMS0441` abgewiesen; Ausnahmen siehe unten | Systembetreuung verständigen; bei Bedarf Meldung `EXC044E` beantworten |
| 5 | Wie bei Sättigungsgrad 4; zusätzlich Einschränkungen bei Systemtasks | Systembetreuung verständigen. |
| 6 (=ZIP) | Wie bei Sättigungsgrad 5; zusätzlich Einschränkung auf Systemlauf, der mit einem Systemstart vom Typ ZIP eingeleitet worden ist | Systembetreuung verständigen. Nachdem wieder genügend freier Speicherplatz auf den Platten geschaffen wurde, neuen Systemlauf mit normalem Typ des Systemstarts einleiten |

Tabelle 26: Sättigungszustand eines Pubsets

Informationen über den Sättigungszustand eines Pubsets liefert auch das Kommando SHOW-PUBSET-SPACE-ALLOCATION.

### Behandlung der Speicherplatzanforderungen

Anforderungen, die von Benutzer-Jobs gestellt werden und deren Erfüllung zum Erreichen des Sättigungsgrades 4 führen würden, werden abgewiesen. Eine Abweichung von diesem Verhalten ergibt sich aus der Belegung des Systemparameters L4MSG.
Ist dieser Systemparameter mit dem Wert 1 belegt, wird für jede Benutzer-Anforderung von Speicherplatz, die zum Erreichen des Sättigungsgrades 4, nicht jedoch auch des Sättigungsgrades 5 führen würde, folgende beantwortbare Meldung an der Bedienstation ausgegeben:

```
EXC044E    SATURATION LEVEL 4 FOR PUBSET/VOLUMESET '(&00)' EXCEEDED.
           REQUEST FOR DISK SPACE ACCEPTED NEVERTHELESS? REPLY (Y=YES; N=NO)
```

Die Bewilligung der Speicherplatzanforderung durch den Operator (Antwort Y) sollte restriktiv gehandhabt werden.

Anforderungen für die Benutzerkennung TSOS werden unabhängig vom erreichten Sättigungsgrad angenommen, sofern nach ihrer Erfüllung noch eine minimale System-Notreserve an Speicherplatz vorhanden ist. Diese Notreserve wird nur bei einem Systemstart im Modus ZIP verbraucht.

Anforderungen von Systemtasks können auch bei Sättigungsgrad 4 oder 5 erfüllt werden.

Anforderungen für die Erstellung von Systemdumps werden abgewiesen, wenn sie zum Erreichen des Sättigungsgrades 5 führen würden.

Bei SM-Pubsets wird versucht, Anforderungen, die sich nicht auf einen bestimmten Volume-Set des Pubsets beziehen, auf dem Volume-Set mit dem niedrigsten Sättigungsgrad zu erfüllen (bei gleich guter Eignung der Volume-Sets in Bezug auf die anderen Kriterien der Volume-Set-Auswahl). Die Anforderung wird abgewiesen, wenn auf allen Volume-Sets, die für die Allokierung in Frage kommen, die Erfüllung der Anforderung zum Erreichen eines Sättigungsgrads (wie oben beschrieben) führen würde.

### Maßnahmen zur Verhinderung der Sättigung

Das Auftreten der Sättigungsgrade 4 und 5 zeigt an, dass der betroffene Pubset/Volume-Set überlastet ist. Wenn ein Pubset bzw. Volume-Set immer wieder in diesen Zustand kommt, ist er nicht ausreichend konfiguriert.

Entsprechende Maßnahmen, die durch die Systembetreuung zu treffen sind, können sein:

- Umverteilung einzelner Benutzerkennungen auf andere Pubsets
- Vergrößerung des Pubsets bzw. Volume-Sets durch Hinzunahme von Platten
- Verdrängung von nicht benutzten Dateien mit Hilfe des Software-Produkts HSMS
- Regelmäßige Aufforderung an die Benutzer, nicht benötigten Speicherplatz freizugeben und Dateien auszulagern
- Kontrolle der Plattenspeicherplatz-Zuweisung mit dem Dienstprogramm SPCCNTRL oder dem Kommando SHOW-PUBSET-SPACE-ALLOCATION
- Sparsamer Umgang mit der Funktion „Speicherplatzüberschreitung“ (Kommando ADD-USER PUBLIC-SPACE-EXCESS=*NO/*ALLOWED)
- Rechtzeitige Reorganisation
- Einsatz von SPACEPRO (siehe Abschnitt „SPACEPRO: Autonome Pubset-Rekonfiguration“ auf Seite 378)

### Festlegen der Sättigungsstufen

Die Einstellung der Schwellwerte für die einzelnen Sättigungsgrade ist Aufgabe der Systembetreuung und erfolgt mit dem Kommando MODIFY-SPACE-SATURATION-LEVEL. Mit diesem Kommando können die Sättigungsstufen sowohl kurzfristig als auch dauerhaft geändert werden (Operand SCOPE=*PERMANENT/*TEMPORARY/*NEXT-PUBSET-SESSION). Für Pubsets bzw. Volume-Sets, für die keine expliziten Festlegungen erfolgt sind, wird die systemglobale Voreinstellung für die Sättigungsstufe 4 wirksam, die mit dem Systemparameter L4SPDEF getroffen worden ist.

Mit dem Kommando MODIFY-SPACE-SATURATION-LEVEL kann auch die System-Notreserve an Plattenspeicherplatz für einen Systemstart im Modus ZIP pubset-spezifisch eingestellt werden (Operand ZIP-LEVEL). Diese wird wie eine 6. Sättigungsstufe behandelt.

Die Einstellungen der Sättigungsstufen können mit dem Kommando SHOW-SPACE-SATURATION-LEVELS für SF-Pubsets bzw. mit Operand VOLUME-SET=*ALL/<cat-id> für die Volume-Sets eines SM-Pubsets abgefragt werden.

Eine eingeschränkte Einstellmöglichkeit für die Sättigungsstufe 4 nur für SF-Pubsets bieten aus Kompatibilitätsgründen auch die Kommandos ADD- bzw. MODIFY-MASTER-CATALOG-ENTRY.

## 9.5.8 Reorganisation von Plattenspeicherplatz

Durch das ständige Anlegen, Löschen, Vergrößern und Verkleinern von Dateien im laufenden Betrieb kommt es auf den Volumes eines Pubsets zu einer immer stärkeren Fragmentierung des freien Speicherbereichs und der anzulegenden Dateien. Die Fragmentierung beeinträchtigt in zunehmenden Maße die Performance der Dateizugriffe und die gleichmäßige Verteilung der I/O-Last über alle Volumes des Pubsets. Die Extent-Liste in den Katalogeinträgen wird durch das zwangsläufige Anlegen mehrerer kleiner Datei-Extents beim Vergrößern der Dateien verlängert.

### SPACEOPT

Das kostenpflichtige Produkt SPACEOPT bereinigt eine Fragmentierung durch die optimale Verlagerung (Reorganisation) der Datei-Extents auf den Volumes eines Pubsets. Ziel von SPACEOPT ist, auf einem Volume möglichst große zusammenhängende freie Speicherbereiche zu schaffen, damit eine Allokierung großer Dateien mit einer geringer Anzahl an Extents erfolgen kann. Des Weiteren wird die Anzahl der Extents nach Möglichkeit verringert, indem mehrere kleine Extents in einem größeren Extent zusammengefasst werden. Im Ergebnis verbessert sich die Performance der Dateizugriffe. Auf den Volumes des Pubsets wird wieder großer, zusammenhängender freier Speicherplatz geschaffen und die Gefahr des Überlaufs der Extent-Listen in den Katalogeinträgen wird beseitigt.

SPACEOPT ist im gleichnamigen Handbuch [54] beschrieben.

### HSMS

Der gemeinschaftliche Plattenspeicher (Pubsets) kann durch eine Sicherung mit anschließender Restauration reorganisiert werden.
Mit folgender HSMS-Anweisung können alle Dateien wieder zusammenhängend zurückgeschrieben und damit der Speicherplatz neu organisiert werden:

```
//RESTORE-FILES REPLACE-FILES-AND-JV=*YES(REORGANIZE-SPACE=*YES)
```

Wenn zusätzlich der Operand RELEASE-UNUSED-SPACE=*YES angegeben wird, wird weiterer Speicherplatz eingespart: Die zugewiesenen (allokierten), aber nicht genutzten Seiten hinter dem Last-Page-Pointer der Datei werden freigegeben.

## 9.5.9 Auskunftsfunktionen

Alle Betriebsparameter und Konfigurationsdaten von Pubsets werden durch entsprechende Auskunftsfunktionen (SHOW-Kommandos) angezeigt, siehe Tabelle 22 auf Seite 307.

Im Detail sind die Kommandos im Handbuch „Kommandos“ [27] beschrieben.

# 9.6 Shared-Pubsets

Als wesentlich im Pubset-Konzept von BS2000/OSD ist der Shared-Pubset zu betrachten. Bis zu 16 Systeme (BS2000/OSD native oder Gastsysteme unter VM2000) können gemeinsam auf ein Pubset zugreifen, sofern alle betroffenen Systeme innerhalb eines MSCF-CCS-Verbundes gekoppelt sind und über direkte Hardware-Pfade zu diesem Pubset verfügen.
Das gesamte Konzept des Shared-Pubset (Hardware-Konfiguration, Verwaltung der Pubsets, Datenzugriffe, Watch-Dog-Mechanismus, Shared-Pubset- und XCS-Verbund) ist ausführlich im Handbuch „HIPLEX MSCF“ [33] beschrieben.



Mehrere BS2000-Systeme unter BS2000/OSD ab V9.0 können gemeinsam auf Net-Storage-Volumes von Shared-Pubsets zugreifen.

## 9.6.1 Shared-Pubset-Verbund

Zur Realisierung des Shared-Pubset-Verbunds wird auf das MASTER/SLAVE-Prinzip zurückgegriffen. Ein System des Verbunds wird zum temporären Eigentümer des Pubsets ernannt („Pubset-Master“) und wickelt zentral alle Verwaltungsfunktionen der Metadaten ab. Alle weiteren am Verbund teilnehmenden Systeme, die Pubset-Slaves oder auch „Slave-Sharer“, richten ihre Verwaltungsanforderungen über MSCF-Funktionen an den Pubset-Master.

Mit dem Kommando SHOW-SHARED-PUBSET können Informationen über den Shared-Pubset ausgegeben werden.

### Systemkennung

Die Systemkennung (Sysid) identifiziert die Systeme in einem Shared-Pubset-Verbund. Beim Shared-Pubset-Betrieb ist darauf zu achten, dass die Systemkennung im Rechnerverbund eindeutig ist.

Bei der Vergabe einer Systemkennung, die intern als Synonym für den BCAM-Namen des Systems geführt wird, muss je nach Pubset-Notation unterschieden werden:

a) Bei einer VSN in PUB-Notation (siehe Seite 316) ist die Systemkennung mit der einstelligen Katalogkennung identisch (Sysid=Catid).

b) Bei einer VSN in Punkt-Notation (siehe Seite 317) kann die Systemkennung einen numerischen Wert von 65 bis 192 annehmen.
Die systemseitige Voreinstellung beträgt 250 (ist also ungültig).

Die Vergabe der Systemkennung erfolgt beim Einrichten des Home-Pubsets mittels SIR über den Operanden SYS-ID der Anweisung DECLARE-PUBSET. Bei existierenden Home-Pubsets kann die Systemkennung über das Kommando SET-PUBSET-ATTRIBUTES neu vergeben werden, die Änderung wird jedoch erst beim nächsten Systemstart wirksam.

### Eigentümer-Auswahl

Im DMS-Record des SVL der Pubres des SF-Pubsets bzw. der Volres des Control-Volume-Sets gibt es zwei Felder, die Auskunft über den gewünschten und den aktuellen Pubset-Master geben. Während der Inbetriebnahme des Shared-Pubsets werden diese Felder ausgewertet und die Eigentümer-Auswahl erfolgt in folgender Reihenfolge:

1. Ist bereits ein aktueller Pubset-Master eingetragen, gilt dieses Fremd-System als Pubset-Master; das eigene System ist dann zwangsläufig ein Pubset-Slave.
2. Bei der Inbetriebnahme selbst wurde mitgeteilt, dass das eigene System Pubset-Master werden soll.
3. Der im SVL als gewünschter Eigentümer eingetragene Pubset wird Pubset-Master.
4. Ist keine der vorangegangenen Bedingungen erfüllt, wird das System Pubset-Master, der als erster die Inbetriebnahme des Shared-Pubsets veranlasst hat.

Das Eintragen des gewünschten Eigentümers eines Shared-Pubsets kann über das Kommando SET-PUBSET-ATTRIBUTES erfolgen. Zusätzlich kann mit Hilfe dieses Kommandos auch ein Backup-Master eingetragen werden, der bei einem eventuellen Master-Ausfall dessen Funktion übernimmt. Diese Vorgehensweise wird Master-Wechsel genannt.
Ist kein Backup-Eigentümer vorgesehen, bleibt der Pubset von den verbliebenen Sharern belegt, ist aber nicht mehr zugreifbar (INACCESSIBLE, QUIET). Der Pubset muss nun entweder an allen Pubset-Slaves exportiert werden oder eines der Systeme wird als Backup-Master zugelassen und führt dann über IMPORT-PUBSET ..., SHARER-TYPE= *MASTER(*MASTER-CHANGE=*YES) einen nachträglichen Master-Wechsel durch.

Das Kommando SHOW-PUBSET-ATTRIBUTES zeigt die Einstellungen des gewünschten Masters, des aktuellen Masters und des Backup-Masters an.

### Verbundaufbau und -abbau

Der Aufbau eines Shared-Pubset-Verbunds geschieht in zwei Schritten:

1. Zunächst wird an allen beteiligten Systemen das Subsystem MSCF gestartet und die erforderlichen Verbindungen werden aufgebaut. Dabei muss mindestens die Verbindung zwischen zukünftigem Pubset-Master und Pubset-Slave bestehen, für einen eventuellen Master-Wechsel zusätzlich die Verbindung zwischen Backup-Master und Pubset-Slave.
2. An allen beteiligten Systemen wird der Shared-Pubset in Betrieb genommen. Die Auswahl des Pubset-Masters erfolgt nach der zuvor beschriebenen Eigentümer-Auswahl. Die Inbetriebnahme an den Pubset-Slaves kann erst dann vollständig abgearbeitet werden, wenn sie am Pubset-Master erfolgreich abgeschlossen ist.

Ein Shared-Pubset-Verbund wird implizit beim Exportieren des Shared-Pubset abgebaut.

### Konfigurationsänderung

Die Konfiguration des Shared-Pubset-Verbunds ist nicht statisch fixiert, sondern kann sich dynamisch ändern. Die Ursachen für eine solche Änderung sind:

- Ein weiterer Pubset-Slave konnektiert sich durch Inbetriebnahme des Shared-Pubsets.
- Ein Pubset-Slave diskonnektiert sich durch Außerbetriebnahme des Shared-Pubsets.
- Ein Pubset-Slave oder der Pubset-Master ist ausgefallen und der Master-Wechsel wurde erfolgreich durchgeführt.

Mit Hilfe des Kommandos SHOW-SHARED-PUBSET kann sich die Systembetreuung einen Überblick über die aktuell gültige Shared-Pubset-Konfiguration verschaffen.

### 9.6.2 Systemüberwachung

Bei Ausfall eines Systems innerhalb des Shared-Pubset-Verbundes müssen die von ihm reservierten Ressourcen freigegeben oder Recovery-Maßnahmen eingeleitet werden. Alle am Shared-Pubset-Verbund beteiligten Systeme werden vom Subsystem MSCF überwacht.

Zur Systemüberwachung dienen zwei voneinander unabhängige Kontrollmechanismen: Zum einen wird zur Realisierung der Lebendüberwachung über alle Sharer-Systeme eines Pubsets die sog. Watch-Dog-Datei ($TSOS.SYS.PVS.SHARER.CONTROL) eingerichtet, in die alle Sharer periodisch Zeitstempel hineinschreiben (Lebendmeldungen). Fällt ein Sharer aus, kann diese Tatsache von einem anderen Sharer am Ausbleiben von dessen Lebendmeldung erkannt und entsprechende Maßnahmen eingeleitet werden. Zum anderen wird beim Ausbleiben der Lebendmeldung die Systemverbindung überprüft, indem ein Auftrag an den betroffenen Sharer gesendet wird; dieser muss innerhalb eines bestimmten Zeitintervalls quittiert werden.
Ein Partner-Ausfall wird nur dann angenommen, wenn das Ausbleiben der Lebendmeldung durch eine erfolglose DFÜ-Überprüfung bestätigt wird.

Bei Ausfall des Eigentümer-Systems wird an allen abhängigen Systemen eine pubset-spezifische Jobvariable gesetzt.

#### Systemausfall im Shared-Pubset-Verbund

Erkennt ein Pubset-Master den Ausfall eines beteiligten Pubset-Slaves, so werden alle vom ausgefallenen Pubset-Slave reservierten Ressourcen freigegeben.

Bei Ausfall des Pubset-Masters findet, angestoßen durch den Watch-Dog-Mechanismus, ein Master-Wechsel statt. Voraussetzung für den Master-Wechsel ist, dass ein aktiver Pubset-Slave im SVL des Shared-Pubsets als Backup-Master eingetragen ist, der die neuen Master-Funktionen übernehmen soll.

Der Backup-Master wird mit dem Kommando SET-PUBSET-ATTRIBUTES BACKUP-MASTER=... im DMS-Record des SVL eingetragen.
Falls kein Backup-Master eingetragen ist oder der eingetragene Backup-Master nicht aktiv ist, entscheidet der Wert des Operanden ALTERNATE-BACKUP, ob der erste aktive Pubset-Slave im SVL zum Pubset-Master wird oder der Operator explizit über das Kommando IMPORT-PUBSET SHARER-TYPE=*MASTER(MASTER-CHANGE=*YES) einen der aktiven Pubset-Slaves zum neuen Pubset-Master bestimmt oder ob der Master-Wechsel mit einem alternativen Backup-Master unterbunden werden soll.

Wenn kein Backup-Master vorgesehen ist oder die Funktion des Master-Wechsels aus einem anderem Grund scheitert, dann ist eine der folgenden Aktionen nötig:

- Alle beteiligten Pubset-Slaves nehmen den Shared-Pubset außer Betrieb und bauen den Shared-Pubset-Verbund komplett neu auf.
- Mit dem Kommando SET-PUBSET-ATTRUBUTES wird die Erlaubnis für einen nachträglichen Master-Wechsel gegeben und dieser mit dem Kommando IMPORT-PUBSET SHARER=*MASTER(MASTER-CHANGE=*YES) angestoßen.

Gründe für ein Scheitern des Master-Wechsels können sein:

- Der eingetragene Backup-Master ist nicht aktiv.
- Eine Verbindung zu einem der beteiligten Pubset-Slaves ist unterbrochen.
- Eines der am Shared-Pubset-Verbund beteiligten Systeme verwendet eine nicht-verbundkompatible Version von HIPLEX MSCF oder einen abweichenden Korrekturstand.

Nach einem erfolgreich durchgeführten Master-Wechsel können alle beteiligten Pubset-Slaves normal weiterarbeiten. Der Master-Wechsel selbst läuft für die Benutzer weitestgehend unerkannt ab.

### 9.6.3 XCS-Verbund

Der XCS-Verbund (Cross-Coupled-System) bietet eine engere Koordination der beteiligten Systeme. Jedes System hat eine konsistente und vollständige Sicht des gesamten Verbunds. Der XCS-Verbund bietet damit Mechanismen zur Realisierung verteilter Anwendungen; er ist in erster Linie als Verfügbarkeits- und Lastverbund von BS2000/OSD konzipiert. Dem Benutzer werden u.a. folgende, im DVS-Umfeld wichtige Funktionen angeboten:

- Distributed-Lock-Manager (DLM)

  Diese Funktion realisiert eine system-übergreifende Sperrenverwaltung und unterstützt damit system-übergreifende Synchronisation und Serialisierung. Sie ist Basisfunktion für SFS.

- Shared-File-System (SFS)

  Das SFS erlaubt innerhalb des XCS-Verbunds die system-übergreifende Aktualisierung von Dateien auf Shared-Pubsets, die nicht notwendig XCS-Pubsets sein müssen. Mit HIPLEX MSCF wird dieser globale Shared-Update für die block- bzw. byte-strom-orientierten Zugriffsmethoden UPAM, FASTPAM und DIV unterstützt.

Ein XCS-Verbund hat weiterreichende Voraussetzungen zu erfüllen als ein Shared-Pubset-Verbund:

- ein System kann max. einem XCS-Verbund angehören
- die Teilnehmer müssen voll vermascht sein, d.h. es müssen MSCF-Verbindungen zwischen allen Systemen des Verbunds bestehen
- dem XCS-Verbund muss mindestens ein XCS-Pubset angehören, zu dem von allen Systemen aus Zugriffspfade vorhanden sein müssen

Ein XCS-Pubset dient als zentraler Ablageort für verbundweit benötigte Daten. XCS-Pubsets werden automatisch durch das System importiert.

Mit dem Kommando SHOW-XCS-PUBSET können Informationen über den XCS-Pubset ausgegeben werden.

Mit dem Kommando SHOW-XCS-OCCUPATION kann angezeigt werden, welche TU-Tasks momentan XCS nutzen.

## 9.7 Erhöhte Datensicherheit durch Standby-Pubsets

Zur Erhöhung der Datensicherheit können Pubsets gespiegelt werden. Plattenspeichersysteme bieten bei Einsatz des Software-Produkts SHC-OSD Schnittstellen zur Erstellung von Standby-Pubsets. Die Standby-Pubset können im Fehlerfall die Funktionen eines ausgefallenen Pubsets übernehmen. Somit werden Ausfallzeiten möglichst klein gehalten und die Verfügbarkeit des Pubsets bzw. des Systems erhöht.

Standby-Pubsets können für Home- und Daten-Pubsets erstellt werden.



Standby-Pubsets dürfen nicht eingesetzt werden, wenn im System die HSMS-Migration eingesetzt wird. Migrierte Dateien sind bei Einsatz eines Standby-Pubsets nicht mehr verfügbar!

Für die Erstellung von Standby-Pubsets sind vor allem die lokalen Replikationsfunktionen der Plattenspeichersysteme empfehlenswert, die mit Kommandos des Software-Produkts SHC-OSD gesteuert werden. Der vorliegende Abschnitt beschreibt, wie Standby-Pubsets mit diesen Funktionen erzeugt und bearbeitet werden.

| **Kommando** | **Bedeutung** |
|---|---|
| ACTIVATE-CLONE [1] | Clone-Paar aktivieren |
| HOLD-MULTI-MIRRORING [2] | Multi-Mirror-Paar trennen |
| MODIFY-IMON-SCI | Pfadnamen im IMON-SCI modifizieren |
| MODIFY-USER-ATTRIBUTES | Eintrag im Benutzerkatalog ändern |
| RESTART-CLONE-SESSION [1] | Clone-Paar wiedererstellen |
| RESTORE-FROM-CLONE [2] | Unit von Clone-Unit rekonstruieren |
| RESUME-MULTI-MIRRORING [2] | Multi-Mirror-Paar rekonstruieren |
| SET-PUBSET-ATTRIBUTES | Charakteristiken eines Pubsets ändern |
| SHOW-CLONE-SESSION-STATUS [1] | Status von Clone-Paaren anzeigen |
| SHOW-MULTI-MIRRORING-STATUS [2] | Status von Multi-Mirror-Paaren anzeigen |
| SHOW-USER-ATTRIBUTES | Einträge des Benutzerkatalogs anzeigen |
| START-CLONE-SESSION [1] | Clone-Paar erstellen |
| START-MULTI-MIRRORING [2] | Multi-Mirror-Paar erstellen |
| STOP-CLONE-SESSION [1] | Clone-Paar auflösen |
| STOP-MULTI-MIRRORING | Multi-Mirror-Paar auflösen |

Tabelle 27: Kommandoübersicht für die Arbeit mit Standby-Pubsets

[1] SHC-OSD-Kommando für alle Plattenspeichersysteme

[2] SHC-OSD-Kommando nur für Symmetrix-Plattenspeichersysteme



Im Rahmen eines Hochverfügbarkeitskonzepts mit mehreren Plattenspeichersystemen an verschiedenen Standorten können Daten-Pubsets auch in das entfernte Plattenspeichersystem gespiegelt werden.

Außerdem können mit der Funktion Snapset, die auf Snap-Funktionen der Plattenspeichersysteme beruht, Pubset-Kopien für Sicherungszwecke erzeugt werden. Zu Snapsets siehe Abschnitt „Datensicherung mit Snapsets“ auf Seite 537. Zu den Replikationsfunktionen der Plattenspeichersysteme siehe Handbuch „SHC-OSD“ [50].

### 9.7.1 Periodische Online-Erstellung eines Standby-Pubsets

Wenn ein Pubset wegen logischer Fehler ausfällt, kommt es zu einem „Crash“. Im Falle eines Home-Pubsets fällt das ganze System aus, bei einem Daten-Pubset laufen (eventuell kritische) Anwendungen nicht weiter. In jedem Fall muss der Betrieb möglichst schnell wieder aufgenommen werden. Deshalb soll ein möglichst aktueller Standby-Pubset (Standby-Home-Pubset oder Standby-Daten Pubset) bereitgehalten werden.

Standby-Pubsets werden mit den Replikations-Funktionen der Plattenspeichersysteme erzeugt. Die Abläufe sind dabei analog. Zu unterscheiden ist, ob mit einem Standby-Pubset oder mit zwei alternierenden Standby-Pubsets gearbeitet wird.

**Verwendung eines Standby-Pubsets**

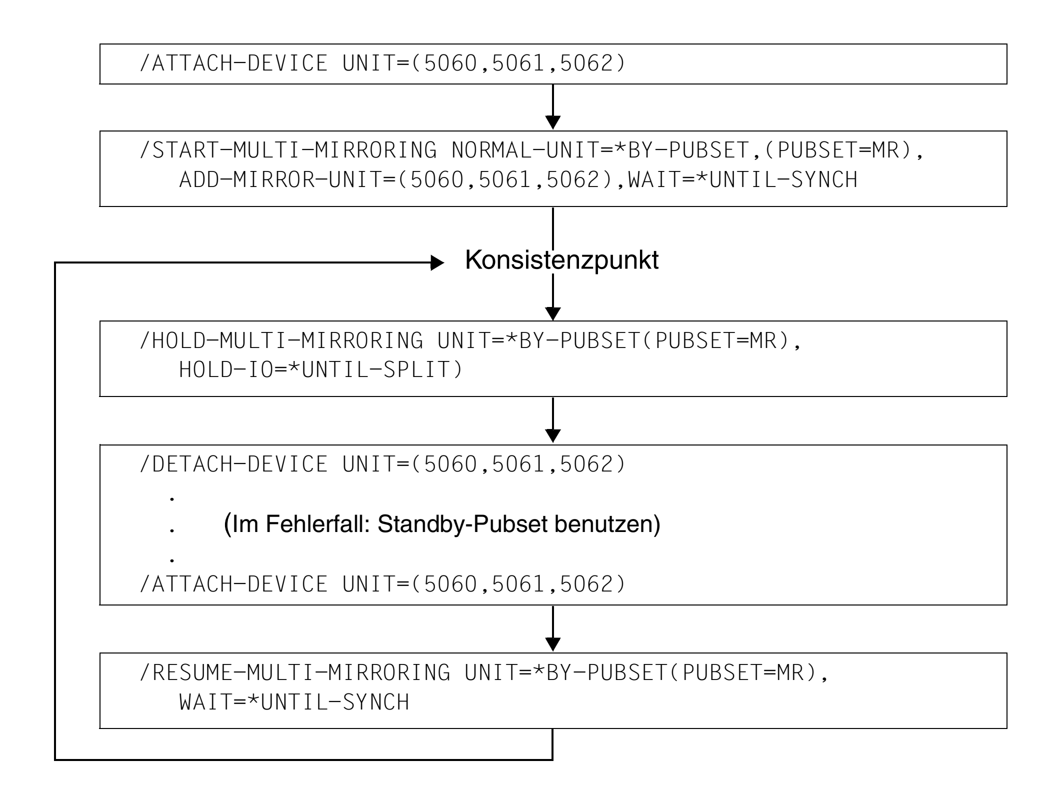
Vorgehensweise:

1. Die Platten zuschalten, die für die Spiegelung benötigt werden
   (Kommando ATTACH-DEVICE).
2. Jeder Platte (Unit) des Pubsets eine Spiegelplatte zuordnen
   (Kommandos START-MULTI-MIRRORING oder START-CLONE-SESSION).
   Dadurch werden die Daten vom Pubset und auf die Spiegelplatten kopiert und dann synchron gehalten.
3. Konsistenzpunkt erstellen.
   - Für Home-Pubsets bedeutet das:
     Host-Puffer für den Home-Pubset zurückschreiben und Anwendungen auf dem Home-Pubset anhalten oder beenden (DAB-Schreib-Caches sind auf dem Home-Pubset nicht erlaubt und müssen deshalb nicht berücksichtigt werden).
   - Für Daten-Pubsets muss eine der folgenden Aktionen ausgeführt werden:
     - Alle Dateien schließen, DAB-Schreib-Caches beenden und Host-Puffer anderer Anwendungen zurückschreiben. Dies kann dadurch erreicht werden, dass alle Anwendungen kurzzeitig beendet werden und der Pubset exportiert wird.
     - Die Anwendungen sorgen selbst für einen Konsistenzpunkt, ohne den Pubset zu exportieren (siehe Handbuch „SHC-OSD“ [50]).
4. Platten und Spiegelplatten trennen
   (Kommandos HOLD-MULTI-MIRRORING oder ACTIVATE-CLONE)
   - Wenn ein Home-Pubset aus mehr als einer Platte besteht, so müssen alle Platten gleichzeitig getrennt werden, um die Datenkonsistenz und Rekonstruierbarkeit zu gewährleisten. Metadaten sollten sich möglichst nur auf einer logischen Platte befinden.

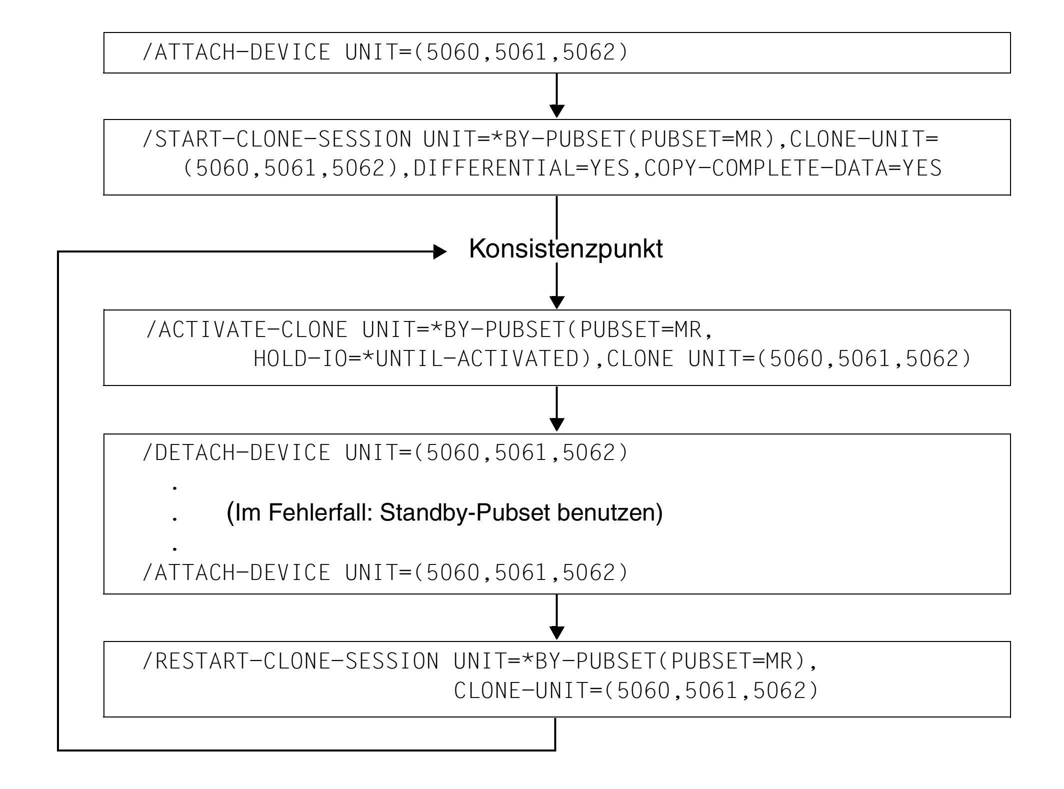
– Wenn ein Daten-Pubset aus mehr als einer Platte besteht, sollten sich Metadaten (Datei-, Benutzer-, Guards-Kataloge usw.) möglichst nur auf einer logischen Platte befinden. Alle Platten müssen zum Splitzeitpunkt gleichzeitig getrennt werden, um die Datenkonsistenz und Rekonstruierbarkeit zu gewährleisten.

5. Die Spiegelplatten wegschalten (Kommando DETACH-DEVICE).
   Dadurch ist der Standby-Pubset erzeugt, der im Fehlerfall genutzt werden kann.
6. Die Platten des Standby-Pubsets wieder zuschalten (Kommando ATTACH-DEVICE).
7. Eine erneute Synchronisation mit dem Pubset veranlassen, damit die Daten des Standby-Pubsets aktualisiert werden
   (Kommandos RESUME-MULTI-MIRRORING oder RESTART-CLONE-SESSION).
8. Weiter bei Schritt 3.

*Beispiel mit TimeFinder/Mirror*



*Beispiel mit lokaler Spiegelung über Clones*



### Verwendung von zwei alternierenden Standby-Pubsets

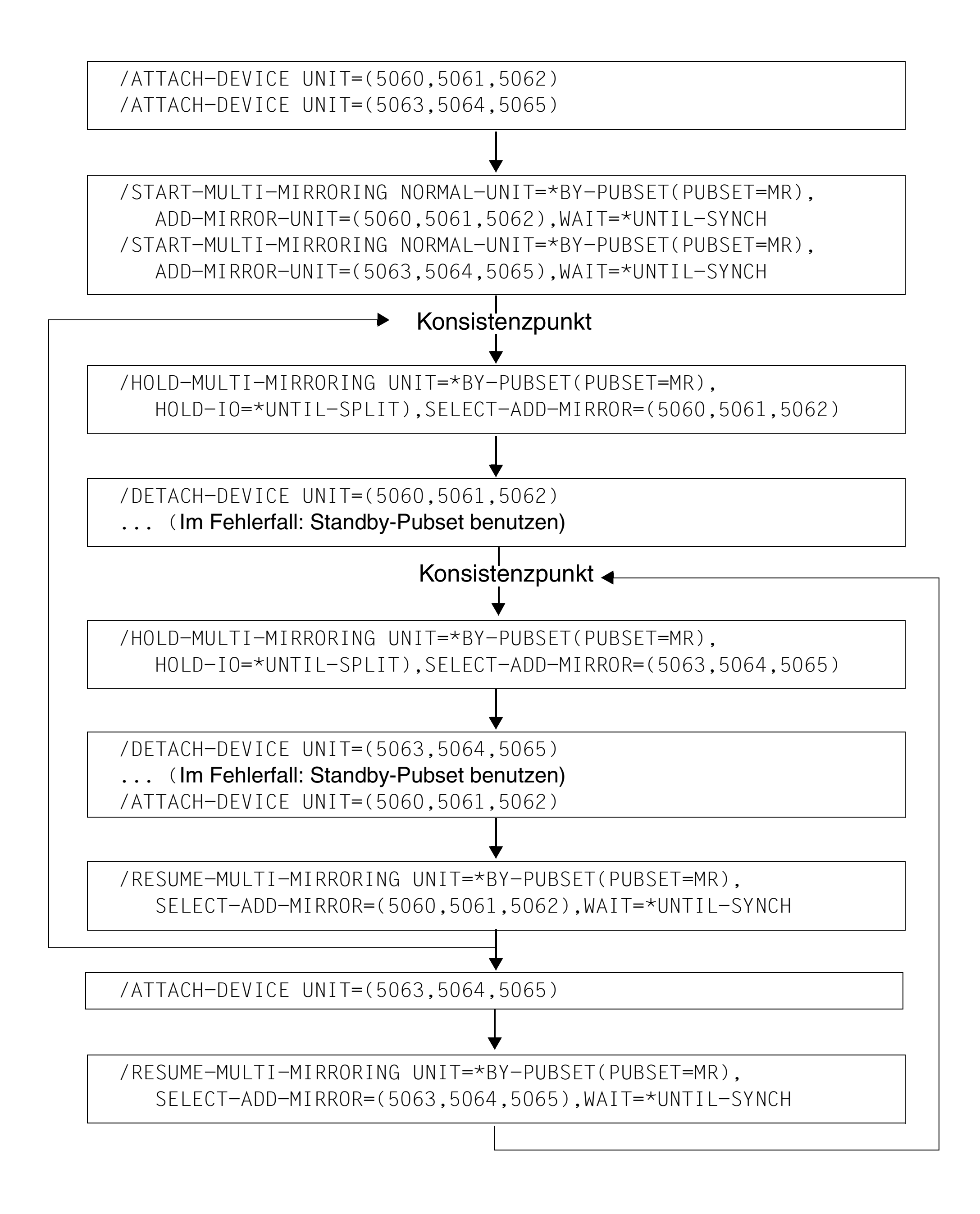
Bei Verwendung von nur einem Standby-Pubset existieren Zeitabschnitte, in denen kein Standby-Pubset zur Verfügung steht (während der Aktualisierung und dem erneuten Abtrennen). Bei kritischen Anwendungen empfiehlt sich deshalb der Einsatz von zwei alternierenden Standby-Pubsets, die abwechselnd bereit stehen und aktualisiert werden.

Vorgehensweise:

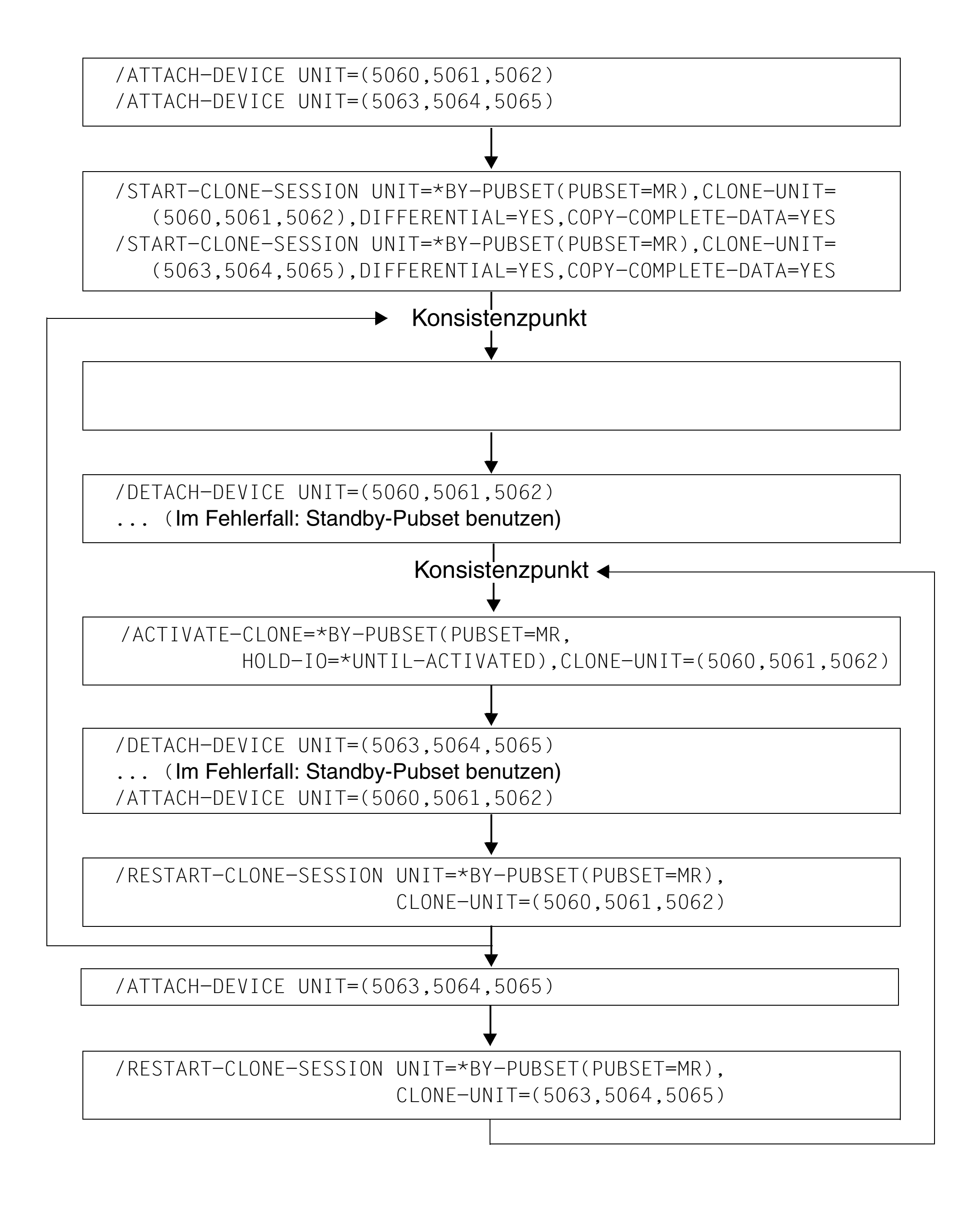
1. Die Platten zuschalten, die für die Spiegelung benötigt werden (Kommando ATTACH-DEVICE).
2. Jeder Platte (Unit) des Pubsets zweimal eine Spiegelplatte (Additional-Mirror-Unit bzw. Clone-Unit) zuordnen (Kommandos START-MULTI-MIRRORING oder START-CLONE-SESSION).
   Dadurch werden die Daten vom Pubset und auf die Spiegelplatten kopiert und dann synchron gehalten.
3. Konsistenzpunkt für den ersten Standby-Pubset erstellen.
   - Für Home-Pubsets bedeutet das:
     Host-Puffer für den Home-Pubset zurückschreiben und Anwendungen auf dem Home-Pubset anhalten oder beenden (DAB-Schreib-Caches sind auf dem Home-Pubset nicht erlaubt und müssen deshalb nicht berücksichtigt werden).
   - Für Daten-Pubsets muss eine der folgenden Aktionen ausgeführt werden:
     - Alle Dateien schließen, DAB-Schreib-Caches beenden und Host-Puffer anderer Anwendungen zurückschreiben. Dies kann dadurch erreicht werden, dass alle Anwendungen kurzzeitig beendet werden und der Pubset exportiert wird.
     - Die Anwendungen sorgen selbst für einen Konsistenzpunkt, ohne den Pubset zu exportieren (siehe Handbuch „SHC-OSD“ [50]).
4. Platten und Spiegelplatten für den ersten Standby-Pubsets trennen (Kommandos HOLD-MULTI-MIRRORING oder ACTIVATE-CLONE-UNIT)
   - Wenn ein Home-Pubset aus mehr als einer Platte besteht, so müssen alle Platten gleichzeitig getrennt werden, um die Datenkonsistenz und Rekonstruierbarkeit zu gewährleisten. Metadaten sollten sich möglichst nur auf einer logischen Platte befinden.
   - Wenn ein Daten-Pubset aus mehr als einer Platte besteht, sollten sich Metadaten (Datei-, Benutzer-, Guards-Kataloge usw.) möglichst nur auf einer logischen Platte befinden. Alle Platten müssen zum Splitzeitpunkt gleichzeitig getrennt werden, um die Datenkonsistenz und Rekonstruierbarkeit zu gewährleisten.
5. Die Spiegelplatten für den ersten Standby-Pubsets wegschalten (Kommando DETACH-DEVICE).
   Dadurch ist der erste Standby-Pubset erzeugt, der im Fehlerfall genutzt werden kann.

6. Einen Konsistenzpunkt für den zweiten Standby-Pubset erzeugen.
7. Platten und Spiegelplatten für den zweiten Standby-Pubsets trennen (Kommandos HOLD-MULTI-MIRRORING oder ACTIVATE-CLONE-UNIT)
8. Die Spiegelplatten für den zweiten Standby-Pubsets wegschalten (Kommando DETACH-DEVICE).
   Dadurch ist der zweite Standby-Pubset erzeugt, der aktuellere Daten als der erste hat und nun im Fehlerfall genutzt werden kann.
9. Die Platten des ersten Standby-Pubsets wieder zuschalten (Kommando ATTACH-DEVICE).
10. Eine erneute Synchronisation des ersten Standby-Pubsets mit dem Pubset veranlassen, damit die Daten des Standby-Pubsets aktuell bleiben (Kommandos RESUME-MULTI-MIRRORING oder RESTART-CLONE-SESSION).
11. Für den ersten Standby-Pubset weiter bei Schritt 3.
12. Wenn der erste Standby-Pubset erneut abgetrennt ist, die Platten des zweiten wieder zuschalten (Kommando ATTACH-DEVICE).
13. Eine erneute Synchronisation des zweiten Standby-Pubsets mit dem Pubset veranlassen (Kommandos RESUME-MULTI-MIRRORING oder RESTART-CLONE-SESSION).
14. Für den zweiten Standby-Pubset weiter bei Schritt 6.

*Beispiel mit TimeFinder/Mirror*



*Beispiel mit lokaler Spiegelung über Clones*



### 9.7.2 Erzeugen eines Home-Pubsets mit neuer Katalogkennung

Innerhalb eines Hochverfügbarkeitsverbundes wird für den Einsatz des Standby-Systems ein eigener Home-Pubset benötigt. Dieser Home-Pubset, der den Charakter eines Standby-Pubsets besitzt, aber eine eigene Katalogkennung hat, kann ebenfalls mit den lokalen Replikationsfunktionen der Plattenspeichersysteme erstellt werden.

Die Erzeugung eines Home-Pubsets mit neuer Katalogkennung wird anhand eines Beispiels dargestellt. Folgende Ausgangssituation ist dabei gegeben:

- Der Pubset MR enthält ein ladefähiges BS2000/OSD der aktuellen Version.
- Der Pubset MR besteht aus drei Platten (Units) mit den mnemotechnischen Gerätenamen 5070, 5071 und 5072.
- Das laufende BS2000-System wurde von dem Pubset MR hochgefahren.
- Für die Spiegelung stehen drei weitere Spiegelplatten zur Verfügung (5073, 5074 und 5075).

**Vorgehensweise**

1. Für den Pubset MR Kopien erzeugen
(Additional-Mirror-Units bzw. Clone-Units 5073, 5074 und 5075):

   – TimeFinder/Mirror:
   ```
   /START-MULTI-MIRRORING NORMAL-UNIT=*BY-PUBSET(PUBSET=MR),
      ADD-MIRROR-UNIT=(5073,5074,5075)
   ```

   – lokale Spiegelung mit Clones:
   ```
   /START-CLONE-SESSION UNIT=*BY-PUBSET(PUBSET=MR),
      CLONE-UNIT=(5073,5074,5075),DIFFERENTIAL=YES,COPY-COMPLETE-DATA=YES
   ```

2. Konsistenzpunkt erstellen. Das bedeutet:
Host-Puffer für den Home-Pubset zurückschreiben und Anwendungen auf dem Home-Pubset anhalten oder beenden (DAB-Schreib-Caches sind auf dem Home-Pubset nicht erlaubt und müssen deshalb nicht berücksichtigt werden).

3. Kopien abtrennen bzw. aktivieren und dabei den Pubset mit der neuen Katalogkennung WG erzeugen.

   – TimeFinder/Mirror:
   ```
   /HOLD-MULTI-MIRRORING UNIT=*BY-PUBSET(PUBSET=MR,NEW-PUBSET=WG,
      HOLD-IO=*UNTIL-SPLIT)
   ```

   – lokale Spiegelung mit Clones:
   ```
   /ACTIVATE-CLONE UNIT=*BY-PUBSET(PUBSET=MR,NEW-PUBSET=WG,
      HOLD-IO=*UNTIL-ACTIVATED)
   ```

Das weitere Vorgehen hängt davon ab, ob das Umbenennen des Pubsets manuell oder mit dem Dienstprogramm PVSREN geschieht.

*Mit Einsatz von PVSREN*

4. PVSREN aufrufen und mit PVSREN-Anweisungen die notwendigen Umbenennungen durchführen:

```
/PVSREN
. . .
//RENAME-PUBSET-OR-VOLUME-SET NAME=MR,NEW-NAME=WG,
  SYSID=173,MODE=*COMPLETE-SHC-RENAME
%  PVR0201 CHANGE DEFAULT CATALOG ID ENTRY 'TT' TO DEFAULT CATALOG ID
ENTRY 'WG' IN USER CATALOG OF HOME PUBSET ? (Y=YES; N=NO)?N
%  PVR0202 CHANGE DEFAULT CATALOG ID ENTRY 'TT' TO DEFAULT CATALOG ID
ENTRY 'WG' IN USER CATALOG OF NEW PUBSET ? (Y=YES; N=NO)?Y
%  PVR0206 CHANGE CATID ENTRY 'TT' IN STANDARD IMON SCI OF HOME PUBSET TO
CATID 'WG' ? (Y=YES; N=NO)?N
% PVR0203 READY PUBSET IN THIS BS2000 SYSTEM ? (Y=YES; N=NO)?N
//END
%  PVR0145 PVSREN TERMINATED NORMALLY
```

Zu PVSREN siehe auch Abschnitt „Erzeugen eigenständiger Pubsets mit PVSREN“ auf Seite 416 bzw. die vollständige Beschreibung im Handbuch „Dienstprogramme“ [15].

*Ohne Einsatz von PVSREN*

4. Dem Pubset WG eine eindeutige Systemkennung geben.
   (Der Pubset WG besitzt noch die Systemkennung (Sysid) des Pubsets MR. Da er selbst als Home-Pubset genutzt werden soll, muss er eine eindeutige Systemkennung erhalten.)
   ```
   /SET-PUBSET-ATTRIBUTES PUBSET=WG,SYSID=173
   ```

5. Pubset WG importieren:
   ```
   /IMPORT-PUBSET PUBSET=WG
   ```

6. Die Pfadnamen im IMON-SCI anpassen:
   ```
   /MODIFY-IMON-SCI SCI-NAME=:WG:$TSOS.IMON.SCI.GPN,
      REFERRED-PUBSET=*PAR(OLD-NAME=MR,NEW-NAME=WG)
   ```

7. Im Benutzerkatalog des Pubsets WG bei allen Benutzereinträgen, die als Standard-Pubset den Pubset MR eingetragen haben, stattdessen den neuen Home-Pubset WG eintragen.
   (Sinnvollerweise wird die Änderung einer größeren Menge von Benutzereinträgen über eine Prozedur erfolgen; bei Einsatz von SDF-P kann z.B. die Variablenausgabe des Kommandos SHOW-USER-ATTRIBUTES genutzt werden).

Bei einer manuellen Vorgehensweise sind dabei die wichtigsten Schritte:

a) Eine Liste erstellen mit allen Benutzerkennungen, die im Benutzerkatalog des Pubsets WG eingetragen sind (z.B. in die Datei LST.USER.WG):

```
/ASSIGN-SYSLST LST.USER.WG
/SHOW-USER-ATTRIBUTES USER-ID=*ALL,INF=*USER-LIST,
   PUBSET=WG,OUTPUT=*SYSLST
/ASSIGN-SYSLST *PRIMARY
```

b) Von jeder der aufgelisteten Benutzerkennungen den Benutzereintrag ausgeben lassen (einschließlich Standard-Pubset der Benutzerkennung) :

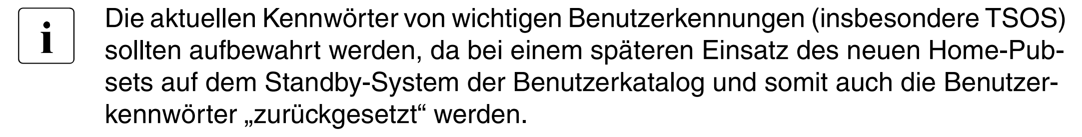
```
/SHOW-USER-ATTRIBUTES USER-ID=<user>,PUBSET=WG
```

c) Wenn für eine Benutzerkennung der Pubset MR als Standard-Pubset eingetragen ist, den Benutzereintrag ändern:

```
/MODIFY-USER-ATTRIBUTES <user>,PUBSET=WG,DEFAULT-PUBSET=WG
```

Insbesondere sind bei dieser Maßnahme die für den Systemlauf wichtigen Benutzerkennungen zu berücksichtigen, z.B.:

- TSOS (wichtige Systemdateien)
- SYSAUDIT (Anlegen der SAT-Logging- und der System-Ereignisstrom-Datei)
- SYSOPR (Aufruf von Run-Dateien; ggf. muss auch die beim Systemparameter NBRUNUID eingetragene Benutzerkennung berücksichtigt werden)
- SYSSPOOL (Start des Subsystems SPOOL)
- SYSFJAM (Filetransfer-Aufträge)
- SYSDUMP und SYSSNAP (Ablage von Systemdumps und SNAP-Dateien)
- SYSHSMS (Einsatz von HSMS)
- Benutzerkennung unter der der MAREN-Katalog abgelegt ist
- Benutzerkennungen für spezifische Anwendungen (z.B. Datenbanken), die beim Zugriff die Standard-Katalogkennung verwenden



Die aktuellen Kennwörter von wichtigen Benutzerkennungen (insbesondere TSOS) sollten aufbewahrt werden, da bei einem späteren Einsatz des neuen Home-Pubsets auf dem Standby-System der Benutzerkatalog und somit auch die Benutzerkennwörter „zurückgesetzt“ werden.

Außerdem ist zu beachten dass innerhalb von Prozeduren und Programmen Dateizugriffe mit expliziter Angabe der Katalogkennung vermieden werden, da diese beim Einsatz des neuen Pubsets zu Fehlern führen.

## 9.8 Erzeugen eigenständiger Pubsets mit PVSREN

Das Dienstprogramm PVSREN kann eigenständige Pubsets aus den Platten-Kopien eines SM- bzw. SF-Pubsets (Pubset-Kopie) innerhalb eines Systems erzeugen.

PVSREN nutzt die lokalen Replikationsfunktionen der Plattenspeichersysteme zum Erstellen von eigenständigen Pubsets, indem Pubset-Kopien umbenannt werden. Bei SM-Pubsets werden die Umbenennungsregeln für die Volume-Sets in einer Parameterdatei abgelegt, in der für jeden Volume-Set durch die Anweisung SET-NAME-OF-NEW-VOLUME-SET die neue Katalogkennung festgelegt wird.

Mit der Anweisung CREATE-PUBSET-FROM-MIRROR wird ein eigenständiger Pubset erstellt. Dabei bestimmt der Operand MIRRORING-METHOD, ob bzw. mit welcher Funktion PVSREN eine Kopie erzeugt:

- Die Angabe von MIRRORING-METHOD=*SPLIT-MIRROR setzt voraus, dass die Pubset-Kopien bereits abgetrennt sind und in Doppelpunkt-Notation (siehe Seite 318) vorliegen. PVSREN führt in diesem Fall nur die Umbenennung zu einem eigenständigen Pubset durch.
- Mit MIRRORING-METHOD=*MULTI-MIRRORING(...)/*CLONE(...)/*SNAP(...) erzeugt PVSREN eine Kopie auf Basis von BCVs, Clone-Units oder Snap-Units und benennt diese zu einem eigenständigen Pubset um.

PVSREN wird ausführlich im Handbuch „Dienstprogramme“ [15] beschrieben.

## 9.9 Erfassen und Beseitigen von Hardware-Fehlern auf Pubsets

Bei Ausfall eines Datenträgers bzw. bei einem partiellen Plattendefekt in einem SF-Pubset oder einem Volume-Set kann mit diesem Pubset/Volume-Set nicht mehr kontinuierlich gearbeitet werden.

Der Fehler kann nur durch eine Neuinitialisierung des betroffenen Datenträgers behoben werden. Damit verbunden ist der Verlust sämtlicher Daten, die sich auf der Platte befinden. Der Pubset kann nur nach einer vollständigen Rekonstruktion aus der Sicherung wieder in Betrieb genommen werden.
Es ist daher notwendig, solche Plattenbereiche rechtzeitig zu erkennen, um sie bei künftigen Speicherplatzanforderungen auszusparen. Nach der Erfassung des Plattenfehlers kann dann eine geeignete Maßnahme zur Beseitigung des Fehlers eingeleitet werden.

### Erfassen von Hardware-Fehlern durch das DVS

Der DVS-Kontrollmechanismus sorgt bei gemeinschaftlichen Datenträgern (Pubsets) dafür, dass permanente Plattenfehler, die während eines Plattenzugriffs durch das DVS auftreten, der Allokierungskomponente bekanntgemacht und von der weiteren Vergabe von Plattenspeicher ausgenommen werden. Es wird damit sichergestellt, dass diese registrierten Plattenbereiche bei künftigen Speicherplatzanforderungen nicht mehr berücksichtigt werden und defekte Plattenbereiche nicht mehr auf andere Dateien übergehen.
Die Informationen über die defekten Plattenbereiche werden in der vom System erzeugten, pubset-spezifischen Defect-Garbage-Datei hinterlegt.
Bei der erstmaligen Registrierung eines Plattenfehlers innerhalb eines Pubsets wird die Datei auf diesem Pubset unter folgendem, vom Pubset-Typ abhängigen Pfadnamen erzeugt:

- bei SF-Pubsets: :<catid>:$TSOS.SYSDAT.DEFECT.GARBAGE.<catid#>
- bei SM-Pubsets: :<catid>:$TSOS.SYSDAT.DEFECT.GARBAGE.<volume-set-id#>

<volume-set-id#> bezeichnet die Katalogkennung des Volume-Sets, auf dem sich die Datei befindet. Ist die Katalogkennung des SF-Pubsets bzw. des Volume-Sets kleiner als vier Zeichen, werden die Variablen <catid#> und <volume-set-id#> auf vier Stellen aufgefüllt.

Die Extentliste dieser Datei besteht aus genau den Extents, die die gemeldeten defekten Plattenblöcke des Pubsets umfassen. Das bedeutet, dass die entsprechenden defekten Blöcke für die Lebensdauer der betroffenen Datei doppelt belegt sind, nämlich einmal in der Datei selbst und einmal in der Defect-Garbage-Datei.

Die Defect-Garbage-Datei darf ausschließlich durch die zuständigen Systemkomponenten bearbeitet werden, jeglicher Zugriff wird abgewiesen. Um einen Zugriff auch aus kleineren BS2000-Versionen zu verhindern, wird ein Dateikennwort vergeben und das Dateischutzattribut ACCESS=READ eingestellt.

Die Defect-Garbage-Datei wird mit dem Backup-Level E versehen, d.h. sie wird nicht durch ARCHIVE oder HSMS gesichert. Damit werden inkonsistente Zustände vermieden.

Die Extentliste der Defect-Garbage-Datei hat – wie bei allen anderen Dateien – nur eine beschränkte Kapazität. Es können, je nach Plattenverteilung, ca. 140 bis 310 Extents aufgenommen werden, was dann der max. Anzahl von fehlerhaften Blöcken pro Pubset entspricht. Bei Überschreitung der Kapazität wird an der Konsole die Meldung `DMS060C` ausgegeben. Tritt dieser Fall ein, muss dringend eine Reorganisation des Pubsets mit der Beseitigung der Plattendefekte (z.B. mit VOLIN) durchgeführt werden.

Eine erzeugte Defect-Garbage-Datei kann ausschließlich durch die IMPORT-PUBSET-Verarbeitung des entsprechenden Pubsets gelöscht werden. Während der IMPORT-PUBSET-Verarbeitung wird geprüft, ob eine Defect-Garbage-Datei vorhanden ist und wenn ja, ob Blöcke daraus bereits repariert wurden. In diesem Fall können die reparierten Blöcke wieder in den normalen Prozess der Plattenspeicherplatzvergabe integriert werden. Das erfolgt implizit in der IMPORT-PUBSET-Verarbeitung über eine F5-Rekonstruktion. Existieren keine (registrierten) defekten Blöcke mehr für diesen Pubset, wird die Defect-Garbage-Datei gelöscht.



**ACHTUNG!**
Die Defect-Garbage-Datei darf in keinem Fall von Hand gelöscht werden; dies muss ausschließlich den Mechanismen des Systems vorbehalten bleiben.
Anderenfalls wären Datenverlust und Zerstörung der F5-Etiketten die Folge!

Bei der F5-Rekonstruktion eines Pubsets während des Import-Vorgangs kann es zu Doppelallokierungen zwischen der Defect-Garbage-Datei und anderen Dateien kommen (Meldungen `DMS0461` und `DMS0462`). Die Meldungen sind jedoch nicht als Hinweis auf Systemfehler oder Inkonsistenzen im F5-Bereich zu verstehen, sie weisen lediglich auf defekte Plattenbereiche im Allokierungsbereich der gemeldeten Dateien hin.

Eine weitere mögliche Fehlermeldung an der Konsole im laufenden Pubset-Betrieb ist `DMS0608`. In diesem Fall wurde gerade ein defekter Block gemeldet. Beim Versuch, ihn in die Defect-Garbage-Datei aufzunehmen, tritt folgender Fehler auf: Es existiert eine Datei mit der zu einer Defect-Garbage-Datei gehörenden Namensstruktur (siehe oben), ihre innere Struktur (Katalogattribute) weicht jedoch von der einer normalen Defect-Garbage-Datei ab. Die Meldung `DMS0608` muss beantwortet werden. Dabei kann versucht werden, die existierende Datei als Defect-Garbage-Datei zu verwenden. Gelingt dies nicht, kann die Funktion abgebrochen werden, d.h. der defekte Block wird nicht erfasst, oder es wird eine neue Defect-Garbage-Datei vom System erzeugt und die existierende implizit gelöscht.

### Manuelle Allokierungssperre

Als zusätzliche Möglichkeit, eine Allokierung auf defekten Platten zu verhindern, kann sowohl für SF- als auch für SM-Pubsets die Allokierung volume-spezifisch und für SM-Pubsets auch volume-set-spezifisch verboten werden (siehe Kommando MODIFY-PUBSET-RESTRICTIONS).

Bereits beim Importieren von SM-Pubsets besteht die Möglichkeit, die Allokierung von Volume-Sets mit defekten Platten zu verhindern. Die Volume-Sets können dabei in den Zustand „defect“ oder „in hold“ versetzt werden (siehe Kommando IMPORT-PUBSET, Operanden DEFECT-VOLUME-SET bzw. IN-HOLD-VOLUME-SET).

### Beseitigen von Plattendefekten

Das Produkt VOLIN lässt zu, bei bestimmten Plattentypen einzelne Blöcke einer Platte zu reparieren, ohne die gesamte Platte neu formatieren zu müssen. Damit sind schnelle Reparaturmaßnahmen für einen Pubset möglich, da die Korrektur auf die betroffenen defekten Blöcke beschränkt ist und Auswirkungen (wie z.B. Datenverlust) nur auf die damit verbundenen Dateien beschränkt bleiben.

Zur Reparatur einer Platte über das Produkt VOLIN ist es notwendig, die Platte exklusiv zur Verfügung zu stellen. Bei SF-Pubsets ist dazu der gesamte Pubset zu exportieren.
Bei SM-Pubsets kann ebenfalls der gesamte Pubset exportiert werden. Zusätzlich gibt es die Möglichkeit, einzelne Volume-Sets (nur bei SM-Pubsets) oder Platten durch dynamische Pubset-Rekonfiguration aus dem laufenden Pubset-Betrieb herauszunehmen (siehe Seite 367). Für die entsprechenden Platten ist dann die Reparatur über VOLIN möglich, der Pubset bleibt in Betrieb. Nach erfolgreicher Korrektur können die Platten bzw. Volume-Sets – ebenfalls über die Schnittstellen der Pubset-Rekonfiguration – wieder in den Pubset integriert werden. Einschränkung: Handelt es sich um die Volres bzw. Pubres, ist dieses Vorgehen nicht möglich.

Zu reparierende Blöcke lassen sich aus der CONSLOG-Datei (siehe „Diagnosehandbuch“ [14]) über die Inserts der Meldung `EXC0857` ermitteln. Der Bezug zu den betroffenen Dateien lässt sich über die Ausgabe des Kommandos SHOW-BLOCK-TO-FILE-ASSIGNMENT herstellen. Bedingung für eine zeitkonsistente Zuordnung der Blöcke zu Dateien ist, dass während und nach der Kommandoausführung keine Katalogisierungs- und Allokierungsvorgänge stattfinden, die die Zuordnungen verändern können.

Nach erfolgreicher Reparatur kann der Pubset wieder importiert werden. Während der IMPORT-Verarbeitung wird eine Prüfung der in der Defect-Garbage-Datei als defekt registrierten Blöcke vorgenommen. Sind diese wieder zugreifbar, werden sie aus der Defect-Garbage-Datei entfernt und der Allokierungskomponente wieder zur normalen Plattenspeicherplatzvergabe zur Verfügung gestellt. Die Markierung der einzelnen Dateien bleibt weiterhin erhalten, da der Block zwar wieder zugreifbar ist, sein Inhalt jedoch noch nicht rekonstruiert wurde (siehe dazu auch Abschnitt „Rekonstruktion von Dateien und Datenträgern“ auf Seite 555).

## 9.10 Verwaltung des SYSEAM-Speicherplatzes

SYSEAM-Dateien nehmen temporäre EAM-Dateien der Benutzer auf.

Temporär bedeutet, dass die EAM-Dateien der Benutzer nur „kurzfristig“ in einer SYSEAM-Datei gespeichert werden. Nach erfolgter Beendigung des Jobs sind die während eines Jobs erzeugten temporären EAM-Dateien wieder gelöscht.

Eine SYSEAM-Datei lässt sich auf jedem Pubset einrichten. Wenn von der Systembetreuung auf dem Standard-Pubset eines Benutzers eine SYSEAM-Datei mit entsprechendem Namen angelegt wurde, so greift der jeweilige Benutzer auf diese SYSEAM-Datei seines Standard-Pubsets zu. Existiert dagegen auf dem Standard-Pubset eines Benutzers keine SYSEAM-Datei, so erfolgt der Zugriff eines Benutzers auf die SYSEAM-Datei seines Home-Pubsets.

Innerhalb eines Shared-Pubset-Verbunds steht abhängig vom Wert des Systemparameters EAMSPVS entweder genau eine SYSEAM-Datei auf dem betroffenen Shared-Pubset (SPVS) für den Pubset-Master des Shared-Pubsets zur Verfügung (EAMSPVS=0, die am Shared-Pubset-Verbund beteiligten Pubset-Slaves können diese SYSEAM-Datei nicht nutzen), oder es kann an jedem der am Shared-Pubset-Verbund beteiligten Systeme (d.h. sowohl am Pubset-Master als auch an jedem Pubset-Slave) mit einer SYSEAM-Datei auf diesem Shared-Pubset gearbeitet werden (EAMSPVS=1).

Die Bestimmungsgrößen der Namensgebung für eine SYSEAM-Datei sind deren Ablageort (Nicht-Shared-Pubset bzw. Shared-Pubset) sowie der Wert des Systemparameters EAMSPVS. Der Name einer SYSEAM-Datei kann daher sein:

- `:<pvsid>:$TSOS.SYSEAM:`
- `:<pvsid>:$TSOS.SYSEAM.<sysid>:`

*Erläuterung*

`:<pvsid>:$TSOS.SYSEAM:`

wenn die SYSEAM-Datei auf einem Nicht-Shared-Pubset liegt, oder, wenn sie auf einem Shared-Pubset liegt und der Systemparameter EAMSPVS den Wert 0 hat

(Anmerkung: In letzterem Fall kann innerhalb eines Shared-Pubset-Verbunds die SYSEAM-Datei auf dem Shared-Pubset ausschließlich vom Pubset-Master des Shared-Pubset verwendet werden, nicht aber von den beteiligten Pubset-Slaves.)

Auf dem durch `<pvsid>` bezeichneten Pubset existiert in diesem Falle genau eine SYSEAM-Datei.

`:<pvsid>:$TSOS.SYSEAM.<sysid>:`

wenn die SYSEAM-Datei auf einem Shared-Pubset liegt und der Systemparameter EAMSPVS den Wert 1 hat (in diesem Fall kann an jedem System des zugrunde liegenden Shared-Pubset-Verbunds, d.h. im Gegensatz zur vorigen Seite sowohl am Pubset-Master als auch an jedem Pubset-Slave, eine SYSEAM-Datei auf dem Shared-Pubset mit dem jeweiligen Namen verwendet werden).

`<sysid>` ist dabei eine 3-stellige Nummer und bezeichnet die Identifikation des importierenden Hosts innerhalb des Shared-Pubset-Verbunds („Systemidentifikation"). Sie kann mit Hilfe des Kommandos SHOW-PUBSET-ATTRIBUTES ermittelt werden.

Auf dem durch `<pvsid>` bezeichneten Pubset existiert somit in diesem Falle für jedes importierende System eine SYSEAM-Datei mit dem Suffix `<sysid>`
(Anmerkung: wenn in diesem Falle von der Systembetreuung eine Datei mit dem Namen `:<pvsid>:$TSOS.SYSEAM` angelegt wird, so wird diese nicht benutzt).

Die Systembetreuung kann für eine SYSEAM-Datei pubset-spezifisch die folgenden Attribute festlegen: Minimal-Größe, Maximal-Größe, Wert für Sekundär-Allokierung sowie Größe eines Cache für die SYSEAM-Datei im Klasse-4-Speicher (dieser wird allerdings ausschließlich für die SYSEAM-Datei auf dem Home-Pubset verwendet). Diese Attribute werden mit Hilfe eines der Kommandos ADD-MASTER-CATALOG-ENTRY bzw. MODIFY-MASTER-CATALOG-ENTRY im betroffenen Master-Catalog-Entry eingetragen.

Wenn im Master-Catalog-Entry eines Pubsets keine Eintragungen für die oben genannten Kenngrößen einer SYSEAM-Datei hinterlegt sind, so werden für deren Minimal-Größe, Wert für Sekundär-Allokierung und Größe des Cache für die SYSEAM-Datei des Home-Pubset globale Standard-Einstellungen wirksam. Diese sind realisiert durch die Systemparameter EAMMIN, EAMSEC und EAMMEM. Darüber hinaus kann der für einen Benutzer allein zur Verfügung stehende Platz innerhalb einer SYSEAM-Datei mit dem Systemparameter EAMSIZ festgelegt werden. Die Behandlung von SYSEAM-Dateien auf einem Shared-Pubset wird durch den Systemparameter EAMSPVS gesteuert (siehe vorige Seite).

Die Werte der genannten Systemparameter können angezeigt werden mit Hilfe des Kommandos SHOW-SYSTEM-PARAMETERS. Sie werden in der Startup-Parameterdatei im Abschnitt mit der Identifikation SYSOPT-CLASS2 festgelegt.
Die Werte für EAMMIN, EAMSEC und EAMSIZ gelten dabei für alle auf einem Pubset erzeugten SYSEAM-Dateien in gleicher Weise. Der Wert für EAMMEM wird nur für eine SYSEAM-Datei auf dem Home-Pubset wirksam, da ein Cache für eine SYSEAM-Datei im Klasse-4-Speicher nur in diesem Fall verwendet wird (genaue Beschreibung der Systemparameter für EAM: siehe die nächsten Seiten oder im Abschnitt „Systemparameter" auf Seite 732).

Eine SYSEAM-Datei wird (spätestens beim ersten Zugriff mit Hilfe der Zugriffsmethode EAM) mindestens in ihrer festgelegten Minimalgröße angelegt; reicht der Platz für eine SYSEAM-Datei nicht mehr aus, so wird die SYSEAM-Datei dynamisch um den Wert für ihre Sekundär-Allokierung erweitert, bis ihre Maximalgröße von 64512 Units erreicht ist.
Bei zurückgehender EAM-Last wird überflüssig gewordener Platz in der SYSEAM-Datei in Stücken des Sekundär-Allokierungs-Wertes wieder dynamisch bis zur unteren Grenze freigegeben.

Die Systembetreuung kann durch Definition des Systemparameters EAMMEM bzw. durch Erzeugen eines entsprechenden Eintrages im Master-Catalog-Entry des Home-Pubsets festlegen, dass bei Systemeinleitung ein Bereich im Klasse-4-Speicher angefordert wird, der im Folgenden als Cache für die SYSEAM-Datei des Home Pubsets genutzt wird.

### Festlegen der Werte für SYSEAM

Das Festlegen der Werte für die Systemparameter von EAM erfolgt in der Startup-Parameterdatei im Abschnitt mit der Identifikation SYSOPT-CLASS2 (beginnend mit dem Record /BEGIN SYSOPT-CLASS2, endend mit dem Record /EOF).

Festlegung:
1 Unit = 3 PAM-Seiten auf Pubsets mit Allocation Unit (AU) = 3 PAM-Seiten
1 Unit = 4 PAM-Seiten auf allen anderen Pubsets

1. Festlegen der Mindestgröße

   EAMMIN = <Anzahl von Units>
   Standard: 3000
   mögliche Werte: 4 - 64512

   EAMMIN sollte so gewählt werden, dass im Normalbetrieb möglichst keine Sekundärzuweisung erforderlich wird.
   Bei SYSEAM-Dateien auf einem Shared-Pubset arbeitet jedes System, das den Shared-Pubset importiert hat, mit seinem eigenen Wert von EAMMIN.

2. Festlegen der Größe für die Sekundär-Zuweisung

   EAMSEC = <Anzahl von Units>
   Standard: 200
   mögliche Werte: 1 - 64512

   EAMSEC sollte so gewählt werden, dass die maximale Größe einer SYSEAM-Datei mit ca. 150 Extents erreicht ist; der Wert sollte ein Vielfaches von 8 betragen.
   Bei SYSEAM-Dateien auf einem Shared-Pubset arbeitet jedes System, das den Shared-Pubset importiert hat, mit seinem eigenen Wert von EAMSEC.

3. Festlegen der pro Benutzer maximal zur Verfügung stehenden Dateigröße

   EAMSIZ = <Anzahl von Units>

   Standard: 64512
   mögliche Werte: 4 - 64512

   EAMSIZ sollte einen Wert von ca. 22000 Units erhalten. Dies ermöglicht einem Benutzer, die maximale Größe einer EAM-Datei (65536 PAM-Seiten) auszunutzen, verhindert jedoch die Überlastung des Systems durch einen Benutzer allein.

   Mit dem Wert EAMSIZ = 64512 (das entspricht der max. Größe der SYSEAM-Datei eines Pubsets) ist es einem einzelnen Benutzer möglich, den gesamten SYSEAM-Speicherplatz des Pubsets zu belegen.

   Bei SYSEAM-Dateien auf einem Shared-Pubset arbeitet jedes System, das den Shared-Pubset importiert hat, mit seinem eigenen Wert von EAMSIZ.

4. Festlegen der Größe für den EAM-Cache im Klasse-4-Speicher

   EAMMEM = <Anzahl von Units>
   Standard: 0
   mögliche Werte: 0 - 2730

   Wenn ein EAM-Cache im Klasse-4-Speicher existiert, werden bei EAM-Aufrufen anstelle von Ein-/Ausgaben auf die Platte nur MVCs durchgeführt.
   EAMMEM gilt ausschließlich für die SYSEAM-Datei auf dem Home-Pubset.

5. Festlegen der Behandlung von SYSEAM-Dateien auf Shared-Pubsets

   EAMSPVS = <Schalter>
   Standard: <Schalter> = 0
   mögliche Werte: <Schalter> = 0 oder <Schalter> = 1

   Der Systemparameter EAMSPVS bestimmt die Namensgebung für SYSEAM-Dateien auf Shared-Pubsets (siehe Seite 420); er wird ausschließlich für SYSEAM-Dateien auf Shared-Pubsets ausgewertet.

*Empfohlene Werte*

EAMMIN = 4500 Units
EAMSEC = 600 Units
EAMSIZ = 20000 Units

## 9.11 SCA: Speed Catalog Access

SCA (Speed Catalog Access) ist ein Software-Produkt zur Beschleunigung der Katalogverwaltung, mit dessen Hilfe die sequenzielle Suche von Katalogeinträgen durch den direkten Zugriff ersetzt wird.

Realisiert wird diese Vorgehensweise mit Hilfe von Tabellen.

Eine Tabelle gibt Aufschluss über freien Platz in den Katalogblöcken (Freelist-Tabelle), die andere ordnet den Datei- bzw. Jobvariablennamen die logische Blocknummer des Katalogblockes zu, in dem der Eintrag steht (Verweistabelle).
Die Freelist-Tabelle wird von Routinen verwaltet, die unter der Aufrufertask ablaufen; die Verweistabelle wird, abhängig von der eingestellten Variante, entweder von der SCA-Task oder von Routinen, die unter der Aufrufertask ablaufen, verwaltet (die möglichen Varianten sind weiter unten erläutert).

SCA darf nur für SF-Pubsets aufgerufen werden. Bei SM-Pubsets ist der beschleunigte Zugriff auf Katalogeinträge bereits im normalen Ablauf implizit enthalten.

### Vorteile beim Einsatz von SCA

*Erhöhung des Gesamtdurchsatzes des Systems*

Durch die geringere Anzahl von Ein-/Ausgabe-Zugriffen zum Dateikatalog und durch die damit verbundenen kürzeren Wartezustände vermindert sich die Systembelastung. Weiterhin werden die gemeinschaftlichen Platten, auf denen sich der Dateikatalog befindet, entlastet. Für das Gesamtsystem bedeutet dies, dass der Durchsatz der Benutzertasks erhöht werden kann.
Die Laufzeit für katalogintensive Programme (z.B. ARCHIVE) sowie das Antwortzeitverhalten bei Kommandos mit hohem Anteil an Katalogzugriffen (z.B. CREATE-FILE, MODIFY-FILE-ATTRIBUTES, IMPORT-FILE) kann erheblich verbessert werden.

*Höhere Flexibilität der Organisation im Data Center*

Organisatorische Maßnahmen zur Verbesserung der Katalogperformance können mit Einsatz von SCA entfallen. Es ist nicht mehr erforderlich, Dateien mit häufigem Katalogzugriff am Anfang einer Benutzerkette zu katalogisieren. Es bestehen somit größere Freiheitsgrade bei der Zuordnung von Dateien zu bestimmten Benutzerkennungen. Probleme beim Übergang auf komplexere Anwendungen lassen sich mit SCA wesentlich reduzieren.

*Keine Kompatibilitätsschwierigkeiten*

SCA verändert die Katalogstruktur nicht. Damit bleibt die Benutzer- wie auch die interne Systemschnittstelle zum Katalogverwaltungssystem CMS erhalten.

Die SCA-Tasks verschiedener Kataloge können gleichzeitig gestartet und beendet werden.

Um die Vorteile von SCA, insbesondere im Hinblick auf die Jobvariablen- und Job-Management-Verarbeitung, bereits während der Startup-Phase nutzen zu können, kann SCA während der Ausführung des Kommandos IMPORT-PUBSET automatisch für diesen Pubset gestartet werden.

SCA wird mit dem Installationsmonitor IMON installiert und benötigt unter der Kennung TSOS folgende Dateien

| | |
|---|---|
| SYSLNK.SCA.180<br>bzw. SKMLNK.SCA.180 | SCA-Modulbibliothek für S-Server<br>SCA-Modulbibliothek für SQ-Server |
| SYSREP.SCA.180 | REP-Datei |
| SPEEDCAT.ENTER.START | Startprozedur |
| SPEEDCAT.ENTER.STOP | Beendigungsprozedur |
| SPEEDCAT.START | Startprogramm |
| SPEEDCAT.STOP | Beendigungsprogramm |

SCA kann für jeden SF-Pubset eingesetzt werden. Folgendes sollte beachtet werden:

- SCA benötigt zum Ablauf zusätzliche Betriebsmittel, wie z.B. virtuellen und realen Speicher, Tasks.
- Wesentliche Performanceverbesserungen sind nur zu erwarten, wenn große Katalogbelegungen der Benutzer vorliegen (> 60-100 Datei-Einträge pro Benutzerkennung).
- Bei optimalem Aufbau der Dateikataloge kann der Einsatz von SCA auf wenige Pubsets reduziert werden.

| **Kommando** | **Bedeutung** |
|---|---|
| ADD-MASTER-CATALOG-ENTRY | Eintrag im Katalogverzeichnis MRSCAT auf dem Home-Pubset erzeugen mit Festlegung, ob SCA bei IMPORT-PUBSET automatisch gestartet werden soll |
| ENTER-JOB | Starten und Beenden von SCA |
| MODIFY-MASTER-CATALOG-ENTRY<br>(EDIT-MASTER-CATALOG-ENTRY) | Eintrag im Katalogverzeichnis MRSCAT auf dem Home-Pubset ändern mit Festlegung, ob SCA bei IMPORT-PUBSET automatisch gestartet werden soll |

Tabelle 28: Kommandoübersicht zu SCA

### Speedcat-Varianten

Beim Starten von SCA für einen bestimmten Pubset kann der Aufrufer angeben, welche Variante des Speedcats geladen werden soll:

- SCA mit Taskwechsel
- SCA ohne Taskwechsel

*SCA mit Taskwechsel*

Zum Starten von SCA wird eine eigene Task, die SCA-Task, aufgebaut. Sie erstellt die SCA-Tabellen und wartet auf Aufträge. Die SCA-Task kommuniziert mit der Benutzertask über eine Zweikammerbörse, um Anforderungen und Ergebnisse auszutauschen.
Mit dieser Variante sind Performance-Einbußen durch den Taskwechsel in Kauf zu nehmen; der Klasse-4-Speicher wird allerdings entlastet, da die SCA-Directories im Klasse-5-Speicher gehalten werden (wo sie nur von der SCA-Task gelesen werden).

*SCA ohne Taskwechsel*

Alle SCA-Aufträge können ohne Taskwechsel durchgeführt werden. Zum Starten von SCA wird auch hier eine eigene SCA-Task aufgebaut, die sich jedoch nach Bereitstellung der Tabellen wieder beendet.
Die Verweistabelle muss hier allen Benutzertasks zugänglich sein und wird zu diesem Zweck im Klasse-4-Speicher angelegt. Der Zugriff wird mittels exklusiver Sperren über eine Signalbörse (Einkammerbörse) reguliert, um zu verhindern, dass mehrere Tasks gleichzeitig lesend und schreibend auf die Directories zugreifen.

### Implizites Starten von SCA bei IMPORT-PUBSET

Wenn SCA bereits bei der IMPORT-Verarbeitung ablauffähig sein soll, dann muss er von der Kommandobearbeitung des IMPORT-PUBSET gestartet werden. Die Information, ob SCA und welche Variante für diesen Pubset automatisch gestartet werden soll, ist im MRSCAT-Eintrag des Pubsets hinterlegt. Mit dem Kommando ADD-MASTER-CATALOG-ENTRY bzw. MODIFY-MASTER-CATALOG-ENTRY kann die Systembetreuung einen entsprechenden Eintrag erzeugen. Der Standardwert für das Starten von SCA im Kommando ADD-MASTER-CATALOG-ENTRY ist START-SPEEDCAT=*AUTOMATIC, d.h. SCA soll automatisch gestartet werden.

### Starten von SCA während des Systemlaufs

1. Start mit Angabe der Katalogkennung:

   ENTER-JOB FROM-FILE=SPEEDCAT.ENTER.START,JOB-NAME=SCA<catid>
   wobei <catid> die ein- oder mehrstellige Katalogkennung des Pubsets ist.
   Die SCA-Variante wird durch Beantwortung der Meldung `DMS03FE` eingestellt.

2. Start ohne Angabe der Katalogkennung:

   ENTER-JOB FROM-FILE=SPEEDCAT.ENTER.START
   wobei die gewünschte Katalogkennung des Pubsets an der Bedienstation erfragt wird.
   Die SCA-Variante wird durch zusätzliche Beantwortung der Meldung `DMS03FE` eingestellt.

### Beenden von SCA

1. Beenden mit Angabe der Katalogkennung:

   ENTER-JOB FROM-FILE=SPEEDCAT.ENTER.STOP,JOB-NAME=SCA<catid>
   wobei <catid> die ein- oder mehrstellige Katalogkennung des Pubsets ist.

2. Beenden ohne Angabe der Katalogkennung:

   ENTER-JOB FROM-FILE=SPEEDCAT.ENTER.STOP
   wobei die Katalogkennung des Pubsets an der Bedienstation erfragt wird.

3. Implizites Beenden:

   Die Kommandos EXPORT-PUBSET und SHUTDOWN beenden die SCA-Task bzw. geben den Klasse-4-Speicher der Verweistabelle und der Freelist frei.

Ohne Angabe der Katalogkennung (2.) können die Kommandos auch vom Operator gegeben werden.
Das Kommando zum Starten von SCA während des Systemlaufs kann auch Bestandteil der CMDFILE sein, und das Kommando zum Beenden von SCA kann in einer RUN-Kommandofolge abgelegt sein.
Dabei sollte beachtet werden, dass SCA erst dann für einen Pubset gestartet werden sollte, wenn der Pubset vollständig importiert ist.

Bei Auftreten eines Fehlers in SCA bzw. bei Hardware-bedingten Schreibfehlern im Katalog schaltet sich SCA automatisch aus und setzt den normalen Katalogzugriff wieder in Kraft. Je nach Art des Fehlers wird ein Systemdump initiiert.

SCA benutzt die SERSLOG-Funktion (siehe „Diagnosehandbuch“ [14]) zur Ausgabe von Diagnoseinformationen. Darüber hinaus werden Fehlermeldungen an der Bedienstation ausgegeben.

# 10 Net-Storage-Verwaltung

BS2000/OSD ab V9.0 ermöglicht den Zugang zu UNIX-Dateisystemen über NFS. BS2000-Dateien können damit im Server-Netzwerk auf dem freigegebenen Speicherplatz von File-Servern abgelegt und bearbeitet werden.

## 10.1 Überblick

In BS2000/OSD gibt es gemeinschaftliche und private Datenträger sowie Net-Storage. Nur Platten können gemeinschaftliche Datenträger sein.
Net-Storage gehört nicht zu den gemeinschaftlichen Datenträgern, dient jedoch zu deren Erweiterung. Net-Storage wird in BS2000/OSD hinsichtlich des Gerätetyps zu der Familie der Platten gerechnet.

Bei der Definition von Platten unterscheidet das DVS gemeinschaftliche Platten (Public Volumes), die zu Pubsets zusammengestellt werden, und private Platten (Private Disks). Net-Storage-Volumes werden als Erweiterung zu Pubsets genutzt.
Einem Pubset können Net-Storage-Volumes zugeordnet werden. Sie sind aber nicht Bestandteil des Pubsets (IMPORT-PUBSET ist auch dann möglich, wenn auf die zugeordneten Net-Storage-Volumes nicht zugegriffen werden kann).

Jeder Pubset enthält einen Benutzerkatalog und seinen eigenen Dateikatalog. Die Angabe NET-STORAGE-USAGE=*ALLOWED im Benutzereintrag legt fest, dass der Benutzer Speicherplatz auf Net-Storage belegen darf.

Alle Datenträger werden in BS2000/OSD durch einen Namen identifiziert, der bis zu sechs Zeichen lang sein darf. Dieser Name wird als VSN (Volume Serial Number) oder auch Archivnummer bezeichnet. Der gültige Zeichenvorrat besteht aus den alphanumerischen Zeichen A..Z, 0..9 und den Sonderzeichen Punkt, @, # und $. Pubset-Volumes sind von anderen Datenträgern durch ihre VSN unterscheidbar, wobei die VSN gemeinschaftlicher Platten einer Konvention unterliegt: Sie muss mit der Zeichenkette „PUB“ beginnen oder einen Punkt an der dritten, vierten oder fünften Stelle enthalten. Diese VSN-Syntax darf für private Datenträger und Net-Storage-Volumes nicht angewandt werden.
Standard-Namen von Net-Storage-Volumes, die einem Pubset zugeordnet sind, haben, abhängig von der Pubset-Notation, eine besondere Form, siehe den Abschnitt „Notation von Net-Storage-Volumes“ auf Seite 432.

Das DVS übernimmt die Speicherplatzverwaltung auf Platten. Einige Kontrollfunktionen (Zugriffsberechtigung, Sättigungssteuerung, usw.) sind nur auf einem Pubset anwendbar, nicht aber auf privaten Platten oder auf Net-Storage.

BS2000-Anwender können Net-Storage-Volumes nach ihrer Zuordnung zu einem lokalen Pubset über die Schnittstellen des DVS nutzen, wenn dies in ihrem Benutzereintrag erlaubt ist. Der verfügbare Speicherplatz auf Net-Storage ist dabei nur durch die Größe des freigegebenen Speicherbereichs auf dem Net-Server begrenzt.
Siehe das Handbuch „Einführung in das DVS“ [19].

## 10.2 Begriffe

Folgende Begriffe werden in BS2000/OSD beim Arbeiten mit Net-Storage verwendet:

**Net-Server**

File-Server im weltweiten Rechnernetz, der Speicherplatz (Network Attached Storage, NAS) für die Nutzung durch andere Server bereitstellt und entsprechende File-Server-Dienste anbietet.

**Net-Storage**

Der von einem Net-Server im Rechnernetz bereitgestellte und zur Nutzung durch fremde Server freigegebene Speicherplatz. Net-Storage kann ein Dateisystem oder auch nur ein Knoten im Dateisystem des Net-Servers sein.

Net-Storage wird mit einem BS2000-System über das Kommando MOUNT-NET-STORAGE verbunden. Der Net-Storage (genauer: das freigegebene Verzeichnis) an sich wird dabei am Net-Client eingehängt. Mit dem Kommando UMOUNT-NET-STORAGE wird der Net-Storage wieder vom BS2000-System getrennt und am Net-Client ausgehängt.

Informationen über den im BS2000-System verfügbaren Net-Storage können mit dem Kommando SHOW-NET-STORAGE angefordert werden.

**Net-Client**

Realisiert den Zugriff auf Net-Storage für das nutzende Betriebssystem.

In BS2000/OSD transformiert der Net-Client zusammen mit dem BS2000-Subsystem ONETSTOR die BS2000-Dateizugriffe in entsprechende UNIX-Dateizugriffe und führt sie über NFS auf dem Net-Server aus.

### Net-Storage-Volume

Net-Storage-Volumes repräsentieren Net-Storage in BS2000/OSD im Rahmen von Daten-Pubsets.

Sie werden mit dem Kommando ADD-NET-STORAGE-VOLUME eingerichtet und einem Pubset zugeordnet. Dabei wird ein Verzeichnis im freigegebenen Dateisystem des Net-Servers eingerichtet und einem lokalen Daten-Pubset (SF- oder SM-) in Form eines Net-Storage-Volumes zugeordnet.

Net-Storage-Volumes werden durch ihre Volume Serial Number (VSN) und den Volumetyp NETSTOR angesprochen. Der Verzeichnisname im freigegebenen Dateisystem des Net-Servers und die VSN des Net-Storage-Volumes sind gleich.

Ein Net-Storage-Volume (genauer: das Verzeichnis mit dem Namen des Net-Storage-Volumes auf dem Net-Server) enthält folgende Dateien:

- ein File-System-Label (Dateiname .FSL) und einen Dateikatalog (Dateiname .BS2FSCAT) mit den Metadaten der auf Net-Storage abgelegten Dateien
- die Benutzerdateien selbst

Informationen über den in einem Daten-Pubset verfügbaren Net-Storage können mit dem Kommando SHOW-PUBSET-NET-STORAGE angefordert werden.

*Notation von Net-Storage-Volumes*

Unabhängig vom Pubset-Typ leitet sich die VSN des Net-Storage-Volumes standardmäßig von dem Pubset-Namen ab:

- Bei einem einstelligen Pubset-Namen wird die PUB-Notation (z.B. `PUBA00`) angenommen und die VSN des Net-Storage-Volumes wird durch Ersetzen des „U“ durch „@“ in der Zeichenfolge „PUB“ gebildet (z.B. `P@BA00`).
- Bei einem mehrstelligen Pubset-Namen wird die Punkt-Notation (z.B. `OTTO.0, ABC.00`) angenommen und die VSN des Net-Storage-Volumes wird durch Ersetzen des Punktes durch „@“ gebildet (z.B. `OTTO@0, ABC@00`).

Mit dieser Notation kann nur das Standard-Net-Storage-Volume im Pubset bezeichnet werden.

*Mehrere Net-Storage-Volumes für einen Pubset*

Wenn einem Pubset mehrere Net-Storage-Bereiche auf einem oder mehreren Net-Servern zugeordnet werden sollen, dann müssen die Net-Storage-Volumes mit unterschiedlichen VSNs zugeordnet werden. Da es pro Pubset nur ein Net-Storage-Volume mit Standard-Namen geben kann, müssen weitere Net-Storage-Volumes dem Pubset mit explizit definierten VSNs (wie bei Privatplatten) zugeordnet werden.

## 10.3 Anbindung von BS2000/OSD an Net-Storage

BS2000/OSD greift auf externe Speichersysteme bisher über Fibre Channel und ein Storage Area Network (SAN) zu.

BS2000/OSD ab V9.0 bietet mit der Bedienung von Net-Storage auch den Zugang zu Net-Servern und dem daran angeschlossenen Network Attached Storage (NAS).

Auf S-Servern erfolgt der Zugriff zu den Net-Servern über den Net-Client HNC.
Auf SQ-Servern wird diese Rolle von X2000 übernommen.

Hinweise zu freigegebenen NAS-Servern finden Sie in den aktuellen Freigabemitteilungen.

## 10.4 Zugriff von BS2000/OSD auf Net-Storage

Der Zugriff auf eine Datei auf Net-Storage durch den BS2000-Anwender erfolgt über DVS-Schnittstellen und den Net-Client auf dem Net-Server in folgenden Schritten:

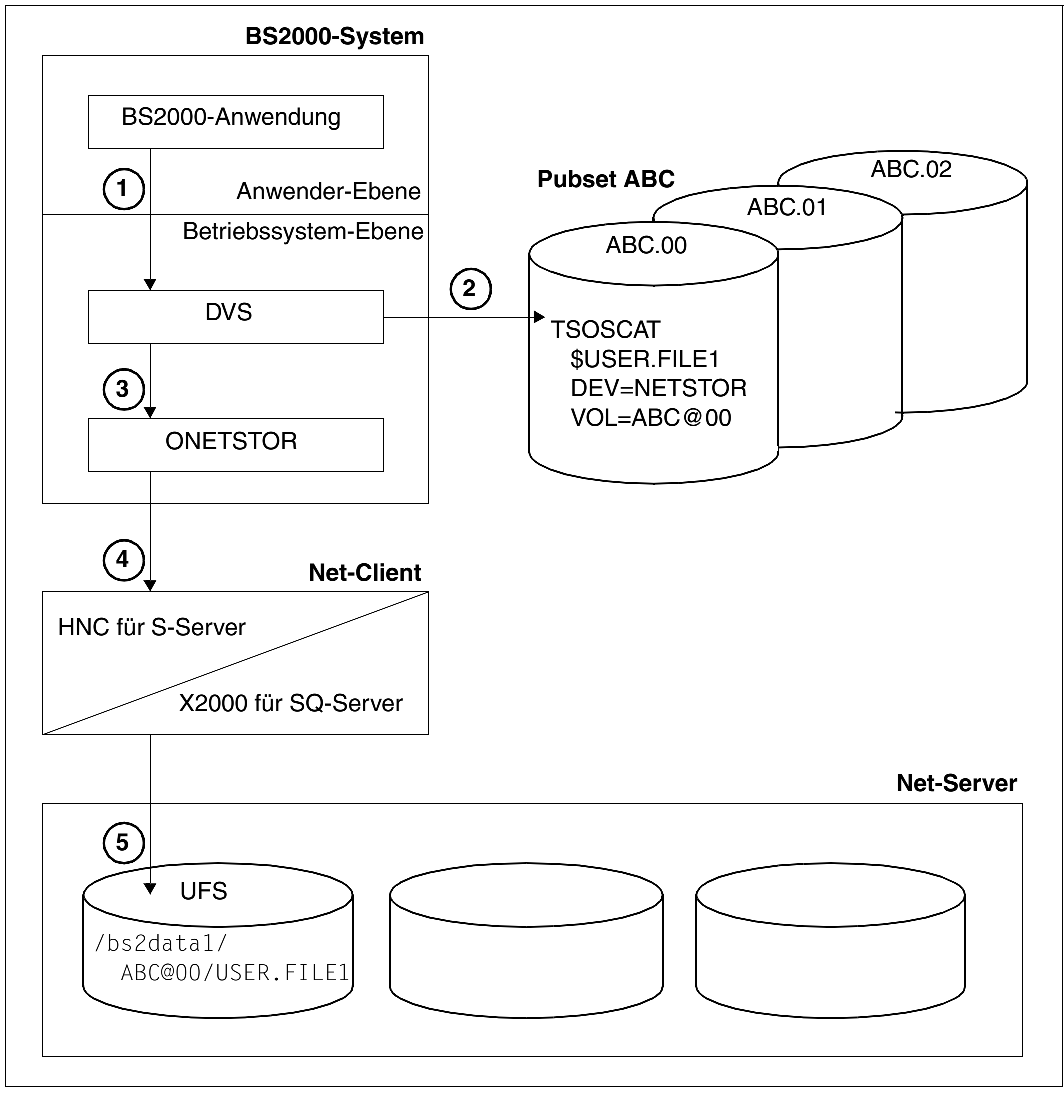



Bild 23: Zugriff auf Net-Storage

1. Eine BS2000-Anwendung unter der Benutzerkennung USER möchte auf die Datei FILE1, die auf Net-Storage liegt und im Pubset ABC katalogisiert ist, zugreifen. Dies geschieht über die üblichen Benutzer-Schnittstellen des DVS.
2. Das DVS prüft die Existenz der Datei im Benutzer- und Dateikatalog des Pubsets ABC. Anhand der Dateiattribute DEVICE=NETSTOR und VOLUME=ABC@00 erkennt das DVS, dass es sich um eine Datei im Verzeichnis ABC@00 auf dem vom Net-Server freigegebenen Net-Storage handelt (siehe den Abschnitt „Net-Storage mit BS2000/OSD verbinden“ auf Seite 437).
3. Der eigentliche Zugriff auf die Datei FILE1 erfolgt über das BS2000-Subsystem ONETSTOR und den Net-Client.
4. Das BS2000-Subsystem ONETSTOR transformiert den BS2000-Dateizugriff in den entsprechenden Dateizugriff im UNIX-Dateisystem (UFS) und reicht ihn an den Net-Client weiter.
5. Am Net-Client (HNC für S-Server, X2000 für SQ-Server) ist das vom Net-Server freigegebenen Verzeichnis `/bs2data1` (mit Unterverzeichnis ABC@00) eingehängt, siehe nachfolgenden Abschnitt „Net-Storage mit BS2000/OSD verbinden“. Der Dateizugriff erfolgt über NFS im UFS des Net-Servers.

## 10.5 Vorbereitungen auf Net-Server und Net-Client

Der Systemverwalter des NFS-Servers richtet eine Freigabe auf dem NFS-Server ein und bestimmt die UserID und GroupID des Benutzers, der die Zugriffsberechtigung auf diese Freigabe erhält. Auf Linux-basierten Systemen wird das freizugebende Verzeichnis in der Regel in die Datei `/etc/exports` eingetragen (Näheres ist in den einschlägigen NFS-How-Tos und Linux/UNIX Man Pages beschrieben), z.B. mit folgender Zeile:

```
/bs2data1  172.10.1.89(rw,nohide,....)
```



Für den BS2000-Zugriff sind stets Schreibrechte notwendig!

Dem freigegebenen Verzeichnis müssen die gewünschte UserID und die GroupID zugeordnet werden (Kommando `chown`). Je nach Hersteller des NFS-Servers können diese Einstellungen auch über spezielle GUIs erfolgen.

Auf dem Net-Client (bei S-Servern auf dem HNC bzw. bei SQ-Servern im X2000) meldet sich der Administrator am HNC- bzw. SQ-Manager an (z.B. unter der Benutzerkennung `sysadm`), ruft das CLI auf und stellt mit dem CLI-Kommando `bs2netsConf` für den Prozess des bs2netsagent dieselbe UserID und GroupID ein, die dem freigegebenen Verzeichnis auf dem NFS-Server zugeordnet wurde.

Da dem bs2netsagent nur eine UserId und GroupId zugeordnet werden kann, müssen auch auf den Net-Servern allen für diesen Net-Client freigebenen Verzeichnissen diese eine UserId und GroupId zugeordnet werden.

Nähere Informationen zu dem Kommando erhalten Sie mit:

– `bs2netsConf` (ohne weitere Optionen) gibt eine Übersicht des Kommandos aus

– `bs2netsConf info [-e]` gibt eine detaillierte Erklärung der Optionen aus (mit -e in Englisch)



Achten Sie bei Wartungsarbeiten (z.B. Software-Update oder Neuinstallation) und Konfigurationsänderungen mit `bs2netsConf` unbedingt darauf, dass Net-Storage, der über den betreffenden Net-Client betrieben wird, vorher mit UMOUNT-NET-STORAGE von allen angeschlossenen BS2000-Systemen getrennt wird.

## 10.6 Net-Storage mit BS2000/OSD verbinden

Um Net-Storage in BS2000/OSD nutzen zu können, muss die Systembetreuung (nach den Vorbereitungen auf Net-Server und Net-Client) zunächst das BS2000-System mit dem Net-Storage verbinden sowie Net-Storage-Volumes einrichten und einem lokalen Pubset zuordnen.

- Verbinden Sie das BS2000-System mit dem Net-Storage, z.B. mit

  ```
  /MOUNT-NET-STORAGE DIRECTORY='/bs2data1',
                     SERVER=*IP-ADDRESS(IP-ADDRESS=172.10.1.110),
                     CLIENT=system1
  ```

  Dieses Kommando führt folgende Bearbeitungsschritte aus:

  – Einhängen des freigegebenen Verzeichnisses am Net-Client

  – Bestandsaufnahme des freigegebenen Verzeichnisses, d.h. alle bereits vorhandenen 6-stelligen Verzeichnisnamen unterhalb dieses Verzeichnisses werden als Volume in die NDM-Tabellen aufgenommen

  – ggf. Belegung des/der Volumes

Für das Einrichten und Zuordnen von Net-Storage-Volumes haben Sie zwei Möglichkeiten:

1. Richten Sie ein neues Net-Storage-Volume ein und weisen Sie es einem lokalen Pubset zu, z.B. mit

   ```
   /ADD-NET-STORAGE-VOLUME VOLUME=*STD, PUBSET=A,
                           DIRECTORY='/bs2data1',
                           SERVER=*IP-ADDRESS(IP-ADDRESS=172.10.1.110)
   ```

   Das Net-Storage-Volume steht nun als Volume mit der Standard-VSN `P@BA00` des angegebenen Pubsets für die Nutzung im BS2000-Betrieb zur Verfügung. Im Verzeichnis `bs2data1` des Net-Storage wird ein neues Unterverzeichnis `P@BA00` angelegt. File-System-Label und Dateikatalog wurden ebenfalls angelegt. Das Net-Storage-Volume kann nun mit Dateien beschrieben werden.

2. Richten Sie ein bestehendes Verzeichnis auf einem Net-Storage-Volume für den BS2000-Betrieb ein, z.B. mit

   ```
   /ADD-NET-STORAGE-VOLUME VOLUME=MYNETS, PUBSET=A,
                           DIRECTORY='/bs2data1',
                           SERVER=*IP-ADDRESS(IP-ADDRESS=172.10.1.110)
                           IMPORT=*YES(OLD-VOLUME-NAME=*SAME)
   ```

   Das Unterverzeichnis `MYNETS` im Verzeichnis `bs2data1` auf dem Net-Server steht nun als Net-Storage-Volume des angegebenen Pubsets für die Nutzung im BS2000-Betrieb zur Verfügung. Das Verzeichnis `MYNETS` enthält noch die Verwaltungsdaten und die darauf eingerichteten BS2000-Dateien. Diese sind aber noch nicht importiert.

Importieren Sie die BS2000-Dateien des Net-Storage-Volumes, z.B. mit

```
/IMPORT-FILE SUPPORT=*DISK(VOLUME=MYNETS, DEVICE-TYPE=NETSTOR,
                           PUBSET=A, FILE-NAME=*ALL)
```

Die Dateien im Verzeichnis MYNETS sind nun im Katalog des Pubsets A eingetragen und können verwendet werden.

## 10.7 Verwalten von Net-Storage in BS2000/OSD

Die Systembetreuung kann Net-Storage über die Schnittstellen des DVS verwalten. Der Net-Storage muss dazu mit dem BS2000-System verbunden sein.

- Net-Storage-Volumes werden im Rahmen der Daten-Pubsets verwaltet.
- Informationen über den im BS2000-System verfügbaren Net-Storage liefert das Kommando SHOW-NET-STORAGE.
- Informationen über die einem Pubset zugeordneten Net-Storage-Volumes liefert das Kommando SHOW-PUBSET-NET-STORAGE.
- Informationen über die den Net-Storage belegenden Tasks des lokalen Systems liefert das Kommando SHOW-NET-STORAGE-OCCUPATION.
- Nachdem der Net-Storage mit dem BS2000-System verbunden ist, können alle Benutzer des Pubsets den Net-Storage nutzen, da im Benutzerkatalog standardmäßig NET-STORAGE-USAGE=*ALLOWED eingestellt ist. Der belegte Speicherplatz auf einem Net-Storage-Volume wird nicht auf das Public-Space-Limit des Benutzers angerechnet.

  Benutzern kann die Erlaubnis, neben den Dateien auf den lokalen Platten des Pubsets auch Dateien auf den Net-Storage-Volumes des Pubsets anzulegen und zu bearbeiten, wieder entzogen werden, z.B. mit

  ```
  /MODIFY-USER-ATTRIBUTES ...,NET-STORAGE-USAGE=*NOT-ALLOWED
  ```

  Das Kommando SHOW-USER-ATTRIBUTES zeigt diese Einstellung.
- Vor dem Einleiten des Shutdown im BS2000-System müssen die Pubsets, denen Net-Storage-Volumes zugeordnet sind, exportiert werden (siehe auch Abschnitt „Net-Storage von BS2000/OSD trennen“ auf Seite 440).

*Rekonfiguration von Net-Storage-Volumes*

Die Systembetreuung kann Net-Storage-Volumes in BS2000/OSD rekonfigurieren, d.h. einem Pubset entziehen und, ohne die Dateien auf dem Net-Storage-Volume zu löschen, einem anderen Pubset zuordnen.

- Entziehen Sie das Net-Storage-Volume dem (importierten) Pubset, z.B. mit

  ```
  /REMOVE-NET-STORAGE-VOLUME ...,FILES-ON-VOLUME=*EXPORT
  ```

  Die Katalogeinträge der Dateien werden nur im lokalen Pubset gelöscht. Die Dateien und Verwaltungsdaten auf dem Net-Storage bleiben bestehen.

- Ordnen Sie das Net-Storage-Volume einem anderen Pubset zu, z.B. mit

  ```
  /ADD-NET-STORAGE-VOLUME ...PUBSET=B,IMPORT=*YES(...)
  ```

  Die Verwaltungsdaten auf dem Net-Storage-Volume werden aktualisiert.

- Erstellen Sie die Katalogeinträge für die Dateien auf dem Net-Storage-Volume im „neuen“ Pubset, z.B. mit

  ```
  /IMPORT-FILE SUPPORT=*DISK(VOLUME=<net-storage-volume>,
                             PUBSET=B,DEVICE-TYPE=NETSTOR,FILE-NAME=*ALL)
  ```

  Die Dateien des Net-Storage-Volumes sind nun im Katalog des Pubsets B eingetragen und können verwendet werden.

- Sie können den Import auch vorab simulieren, z.B. mit:

  ```
  /CHECK-IMPORT-DISK-FILE VOLUME=P@BA00,DEVICE-TYPE=NETSTOR,...
  ```

- In bestimmten Fällen, z.B. zur Beseitigung von Inkonsistenzen nach Ausfall eines Net-Storage, kann es sinnvoll sein, einzelne Dateien vom Pubset zu exportieren:
  Hierzu muss beim EXPORT-FILE ein Net-Storage-Volume oder der Storage-Typ *NET-STORAGE angegeben werden. (Andernfalls wirkt EXPORT-FILE nur auf Dateien auf privaten Datenträgern.)

  ```
  /EXPORT-FILE ...,SELECT=*BY-ATTRIBUTES(STORAGE-TYPE=*NET-STORAGE,...)
  ```

  Entsprechend können Sie Dateien, die in BS2000/OSD noch nicht bekannt sind, importieren, d.h. einen Katalogeintrag im lokalen Pubset erstellen, z.B. mit:

  ```
  /IMPORT-FILE SUPPORT=*DISK(VOLUME=P@BA00,DEVICE-TYPE=NETSTOR,
                               FILE-NAME= ... )
  ```

## 10.8 Net-Storage von BS2000/OSD trennen

Net-Storage kann mit UMOUNT-NET-STORAGE vom BS2000-System getrennt werden. Laufende Anwendungen können dann nicht mehr auf Dateien, die auf Net-Storage liegen, zugreifen.

Vor dem Einleiten des Shutdown im BS2000-System müssen die Daten-Pubsets, denen Net-Storage-Volumes zugeordnet sind, exportiert werden. Damit wird sichergestellt, dass auf dem Net-Server keine geöffneten Dateien mehr existieren.

BCAM darf erst nach dem Exportieren des Pubsets beendet werden. Wenn BCAM vorher beendet wird, können geöffnete Dateien auf Net-Storage (u. A. auch der Katalog) nicht mehr geschlossen und die Verbindung zwischen Net-Client und Net-Server nicht abgebaut werden. In diesem Fall ist der Zugriff zu dem Net-Storage-Volume bei einem erneuten Pubset-Import erst wieder möglich, wenn die Meldung `DMS1326` beantwortet wird.

Net-Storage kann auch im laufenden BS2000-Betrieb getrennt werden, z.B. bei einem Wechsel des Net-Servers:

- Trennen Sie den Net-Storage vom BS2000-System, z.B. mit

  ```
  /UMOUNT-NET-STORAGE DIRECTORY='/bs2data1',
                      SERVER=*IP-ADDRESS(IP-ADDRESS=172.10.1.110),
                      CLIENT=system1
  ```

  Das Kommando wird nur ausgeführt, wenn keine Belegung des Net-Storage mehr vorliegt. Belegungen durch das lokale System können Sie mit dem Kommando SHOW-NET-STORAGE-OCCUPATION überprüfen. SHOW-NET-STORAGE-OCCUPATION zeigt alle belegenden Tasks des lokalen Systems für eine Mount-Id an. Auch für Shared-Pubsets werden nur die belegenden Tasks des lokalen Systems ausgegeben.

  Im Notfall können Sie die Ausführung des Kommandos UMOUNT-NET-STORAGE mit der Option FORCE=*YES erzwingen: Der Net-Storage wird auch dann getrennt, wenn es noch belegende Tasks gibt. Dabei können jedoch Inkonsistenzen entstehen, da geöffnete Dateien nicht ordnungsgemäß geschlossen werden.

Mit dem Kommando UMOUNT-NET-STORAGE hängen Sie auch das freigegebene Verzeichnis am Net-Client aus.

## 10.9 Randbedingungen

Für das Arbeiten mit Net-Storage sind folgende Randbedingungen zu beachten:

- Net-Storage kann nur einem Daten-Pubset zugeordnet werden. Dem Home-Pubset kann kein Net-Storage zugeordnet werden.
- Eine Datei auf Net-Storage besteht nur aus einem Extent. Die Angabe einer Volume-Liste wird akzeptiert, es wird aber nur das erste Volume verwendet.
- Eine Speicherplatzbelegung im Operanden SPACE der Kommandos CREATE-FILE und MODIFY-FILE-ATTRIBUTES kann zwar angegeben werden, es wird aber kein Speicherplatz in dieser Größe auf Net-Storage belegt. Die (maximale) Speicherplatzbelegung findet erst dann statt, wenn die Datei in der entsprechenden Größe gespeichert wird.
- Ein Net-Storage-Volume verhält sich in Bezug auf die Allokierung wie eine NK2-Platte mit minimaler Allokierungseinheit 8 KB.
- Absolute Allokierung ist für Dateien auf Net-Storage nicht möglich.
- Dateien mit folgenden Eigenschaften können **nicht** auf einem Net-Storage-Volume eingerichtet werden:
  - Dateien mit PAM-Key
  - Arbeitsdateien
  - temporäre Dateien
  - Dateigenerationsgruppen
- Ein SMPGEN-Lauf wird abgebrochen, wenn Pubsets mit Dateien auf Net-Storage bearbeitet werden. Katalogeinträge von Dateien auf Net-Storage müssen auf den betroffenen Pubsets vor dem SMPGEN-Lauf entfernt oder exportiert werden, z.B. mit

  ```
  /REMOVE-NET-STORAGE-VOLUME ...,FILES-ON-VOLUME=*EXPORT
  ```
- Wenn ein Pubset, dem Net-Storage-Volumes zugeordnet sind, mit PVSREN umbenannt wird, wird auch ein evtl. vorhandenes Volume mit Standard-Namen (P@BX00, XY@000,...) entsprechend umbenannt. Net-Storage-Volumes mit benutzerdefinierten Namen werden nicht umbenannt.

- Der Administrator des File-Servers kann für Anwendungen, die auf unterschiedlichen Betriebssystemen ablaufen, den direkten Zugriff auf gemeinsame Dateien auf Net-Storage zulassen. Hierbei ist Folgendes zu beachten:
  - Anwendungen fremder Betriebssysteme (nicht BS2000/OSD) dürfen nur lesend auf BS2000-Dateien zugreifen.
  - Zugriffsschutz in BS2000/OSD hat nur dann eine Wirkung, wenn der Zugriff von BS2000/OSD aus erfolgt. Sonst hat er keine Wirkung.

### Shared-Pubsets

Mehrere BS2000-Systeme unter BS2000/OSD V9.0 können gemeinsam auf Net-Storage-Volumes von Shared-Pubsets zugreifen. Es wird empfohlen, an Pubsets, die Katalogeinträge von Dateien auf Net-Storage enthalten, nur Sharer mit BS2000/OSD ab V9.0 zu betreiben. Systeme mit BS2000/OSD < V9.0 können nicht auf Net-Storage-Dateien zugreifen.

Alle Sharer müssen sich mit dem entsprechenden Net-Storage verbinden (MOUNT-NET-STORAGE).

Im Shared-Pubset-Verbund kann ein neues Net-Storage-Volume nur am Master-Rechner angelegt oder auch gelöscht werden. Wurde ein Net-Storage-Volume am Master angelegt, muss dieses Net-Storage-Volume auch an allen Slaves jeweils über das Kommando ADD-NET-STORAGE-VOLUME bekannt gemacht und belegt werden. Erst danach kann auch vom Slave-Rechner auf das Net-Storage-Volume zugegriffen werden (Status IN-USE). Diese Belegung bleibt bis zum Export des Pubsets erhalten.

Danach ist auch bei erneutem IMPORT-PUBSET an einem Slave-Rechner das Net-Storage-Volume sofort verfügbar (existierende Verbindung zum Net-Storage vorausgesetzt).

Bei einem Masterwechsel aktualisiert das System auf den zugehörigen Net-Storage-Volumes die eingetragene Master-SYSID. Dies ist aber nur für verfügbare Net-Storage-Volumes (Status: IN USE) möglich. Sollte bei einem Master-Wechsel ein Net-Storage-Volume nicht verfügbar gewesen sein (nicht verbunden oder nicht belegt (Status: ONLINE)), kommt es beim ersten Zugriff auf das Volume zur Meldung DMS1326, die an der Konsole des Master-Rechners ausgegeben wird und beantwortet werden muss.

### Platten-Spiegelung

Pubsets, auf denen Net-Storage-Dateien katalogisiert sind, können mit DRV oder SHC-OSD gespiegelt werden. Allerdings werden die Net-Storage-Dateien selbst nicht gespiegelt, da sie auf einem anderen Medium gespeichert sind. Nach Auflösen der Spiegelung sind somit Dateien auf Net-Storage über einen abgetrennten Spiegel nicht zu erreichen.

Es sind folgende Fälle zu unterscheiden:

- Abtrennen des Spiegels ohne Umbenennen

  Der Original-Pubset wird wieder importiert. Net-Storage-Dateien können weiter wie gewohnt über den Original-Pubset angelegt, gelöscht oder bearbeitet werden.

  Wenn der abgetrennte Spiegel wieder mit dem Original-Pubset synchronisiert wird, werden implizit auch die Katalogeinträge der Net-Storage-Dateien aktualisiert.

  Zu beachten ist, dass im Rahmen des Auftrennens der Spiegel keine Änderungen an den Verwaltungseinträgen der Net-Storage-Volumes vorgenommen werden. D.h. der Pubset, der als erster wieder importiert wird, erhält Zugriff zum betreffenden Net-Storage (existierende Verbindung zu Net-Storage vorausgesetzt).

  Wenn der abgetrennte Spiegel an einem anderen System importiert wird, wird bei Zugriff auf die zugeordneten Net-Storage-Volumes die beantwortbare Meldung DMS1326 ausgegeben, da das Volume eine fremde SYSID enthält. Wenn die Meldunge mit „ja“ beantwortet wird, dann wird der Zugriff auf das Volume erlaubt. Zu beachten ist, dass falls das Volume tatsächlich an einem anderen System belegt ist, es durch die falsche Beantwortung der Frage zu Inkonsistenzen und zu Datenverlust kommen kann.

  Ist eine derartige Weiterverarbeitung des abgetrennten Spiegels geplant, wird folgendes Vorgehen empfohlen:

  a) Entziehen der dem Pubset zugeordneten Net-Storage-Volumes:

     ```
     /REMOVE-NET-STORAGE-VOLUME ..., FILES-ON-VOLUME=*EXPORT
     ```

  b) Den Pubset exportieren und den Spiegel auftrennen.

  c) Gewünschten Pubset, Original oder Spiegel, importieren.

  d) Net-Storage-Volumes wieder dem importierten/aktiven Pubset zuordnen:

     ```
     /ADD-NET-STORAGE-VOLUME ..., IMPORT=*YES
     ```

  e) Die Katalogeinträge der Dateien des Net-Storage-Volumes mit IMPORT-FILE wieder in den TSOSCAT des importierten Pubsets aufnehmen.

  f) Der abgetrennte Spiegel enthält nun keine Net-Storage-Dateien mehr und kann an einem anderen System weiterverarbeitet werden.

- Abtrennen des Spiegels mit Umbenennen

  Es wird empfohlen die Net-Storage-Volumes vor dem Auftrennen des Spiegels vom dem Pubset zu entziehen:

  ```
  /REMOVE-NET-STORAGE-VOLUME ..., FILES-ON-VOLUME=*EXPORT
  ```

  Nach dem Abtrennen des Spiegels kann der Net-Storage wieder dem gewünschten Pubset-Spiegel zugeordnet werden:

  ```
  /ADD-NET-STORAGE-VOLUME ..., IMPORT=*YES
  ```

  Danach müssen die Dateien vom Net-Storage-Volume importiert werden.

  Wenn gewünscht wird, dass die Dateien auf Net-Storage permanent verfügbar sind, dann kann alternativ auch folgendermaßen vorgegangen werden:

  a) Der Spiegel wird mittels DRV oder SHC-OSD Anweisung abgetrennt und dabei umbenannt. Beim Abtrennen des Spiegels werden auf diesem die Net-Storage-Volume Einträge implizit für ungültig erklärt.

  b) Beim Import dieses Pubsets werden die Net-Storage-Volume-Einträge automatisch gelöscht.

  c) Nach Import des Pubsets muss der Systemverwalter die verbliebenen und nun ungültigen Katalogeinträge der Net-Storage-Dateien exportieren:

  ```
  /EXPORT-FILE ..., DEVICE=NETSTOR
  ```

  d) Nun kann der Pubset entweder ohne Net-Storage weiterverwendet werden oder es können neue Net-Storage-Volumes angelegt werden. Es können auch vorhandene Net-Storage-Volumes wieder in Betrieb genommen werden:

  ```
  /ADD-NET-STORAGE-VOLUME ..., IMPORT=*YES
  ```

  e) Sollen die Net-Storage-Dateien und -Volumes des Original-Pubsets dem abgetrennten und bereinigten Spiegel hinzugefügt werden, müssen diese vorher vom Original exportiert werden:

  ```
  /REMOVE-NET-STORAGE FILES-ON-VOLUME=*EXPORT
  ```

# 11 Job- und Task-Steuerung

In diesem Kapitel werden die Job-Steuerung, die alle wartenden Aufträge (Jobs) verwaltet, und die Task-Steuerung, die alle laufenden Aufträge (Tasks) verwaltet, beschrieben.
Das Programm JMP (Jobpool Management Program) rekonstruiert aus den Informationen, die das JMS über akzeptierte Batch-Jobs im Job-Pool hinterlegt hat, ENTER-JOB-Kommandos.
Im Abschnitt „PCS: Performance-Überwachungssystem“ auf Seite 494 werden die Zusammenhänge zwischen Job- und Task-Steuerung und dem Performance-Überwachungssystem PCS kurz dargestellt.

Es lassen sich grundsätzlich folgende Benutzer-Jobs unterscheiden:

- Dialog-Jobs
- Batch-Jobs (ENTER-Job oder SPOOLIN)
- SPOOLOUT-Jobs

Mit Ausnahme der SPOOLOUT-Jobs beginnen alle Benutzer-Jobs mit dem Kommando SET-LOGON-PARAMETERS und enden mit dem Kommando EXIT-JOB.
SPOOLOUT-Jobs werden entweder mit dem Kommando PRINT-DOCUMENT eingeleitet oder nach Task-Ende zur Ausgabe bereitgestellt.

Ein Job wird von der Job-Steuerung einer Job-Klasse zugeordnet und in die entsprechende Warteschlange eingereiht. Jeder Job erhält sofort bei Job-Annahme eine Auftragsnummer (TSN), mit deren Hilfe er während seiner Verweilzeit im System angesprochen werden kann. Die Job-Steuerung steuert und verwaltet alle Jobs.

Ein Job wird zur Task, wenn ihm Systemressourcen (CPU, Speicher, Geräte) zugeteilt werden. Die Task-Steuerung richtet einen Task-Control-Block (TCB) ein.
Die Task-Steuerung steuert und verwaltet alle Tasks.

Das Job-Management-System (JMS) sorgt für die Annahme und Abwicklung der Benutzer-Jobs. Die wesentlichen Aufgaben des JMS sind:

- Auftragsannahme mit Überprüfung der Zugangsberechtigung
- Job-Auswahl zur Startfreigabe
- Job-Initialisierung

Diese Aufgaben werden durch die Komponenten der Job-Steuerung (Job-Klassen und Job-Streams) realisiert.

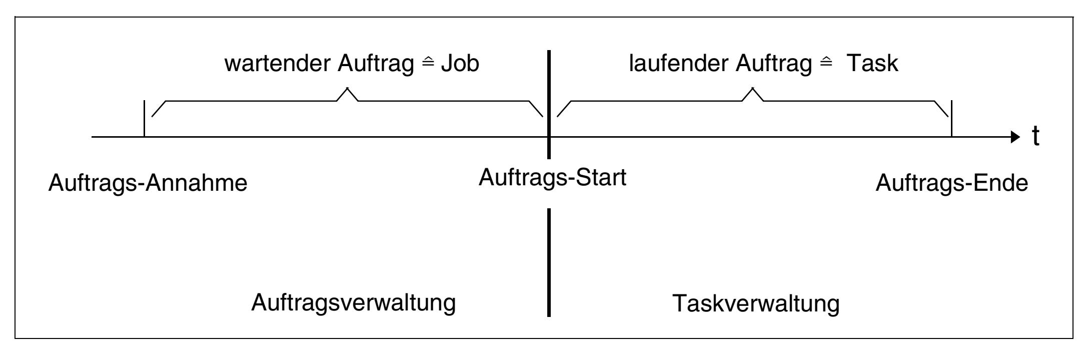



Bild 24: Zusammenhang Job-Task

## 11.1 Job-Steuerung

Die Job-Steuerung (Auftragssteuerung) ermöglicht eine differenzierte und damit rationellere Steuerung von Jobs. Dabei sind vor allem zwei Eigenschaften der Job-Steuerung von Bedeutung:

*Die Jobs können in Job-Klassen eingeteilt werden.*

Job-Klassen beschreiben ein Job-Profil und sind ein Ordnungsbegriff für Jobs, die in bestimmten Eigenschaften und Merkmalen übereinstimmen.

*Die Job-Klassen werden Job-Schedulern zugeordnet.*

Die Job-Klassen werden von der Systembetreuung max. 16 Job-Schedulern zugeordnet, wobei die Job-Scheduler unterschiedliche Strategien verfolgen können und entscheiden, welcher Job zum Start freigegeben wird.
Die System-Job-Klasse $SYSJC ist fest dem System-Job-Scheduler $SYSJS zugeordnet.

Im Normalfall werden alle von den Job-Schedulern zum Start freigegebenen Jobs vom Klassen-Scheduler sofort der Task-Steuerung zum Start übergeben.

Nach Aufhebung einer Überlastsituation oder einer Speichersättigung werden die Jobs der Job-Klasse bevorzugt, deren Klassenoptimum noch nicht erreicht ist und die die höchste Dringlichkeit besitzt (abhängig von der Anzahl bereits in der Job-Klasse laufender Jobs und dem Klassengewicht). Der Klassen-Scheduler bildet die Schnittstelle zur Task-Steuerung. Er fällt die Entscheidung, ob ein zum Start freigegebener Job tatsächlich abläuft.

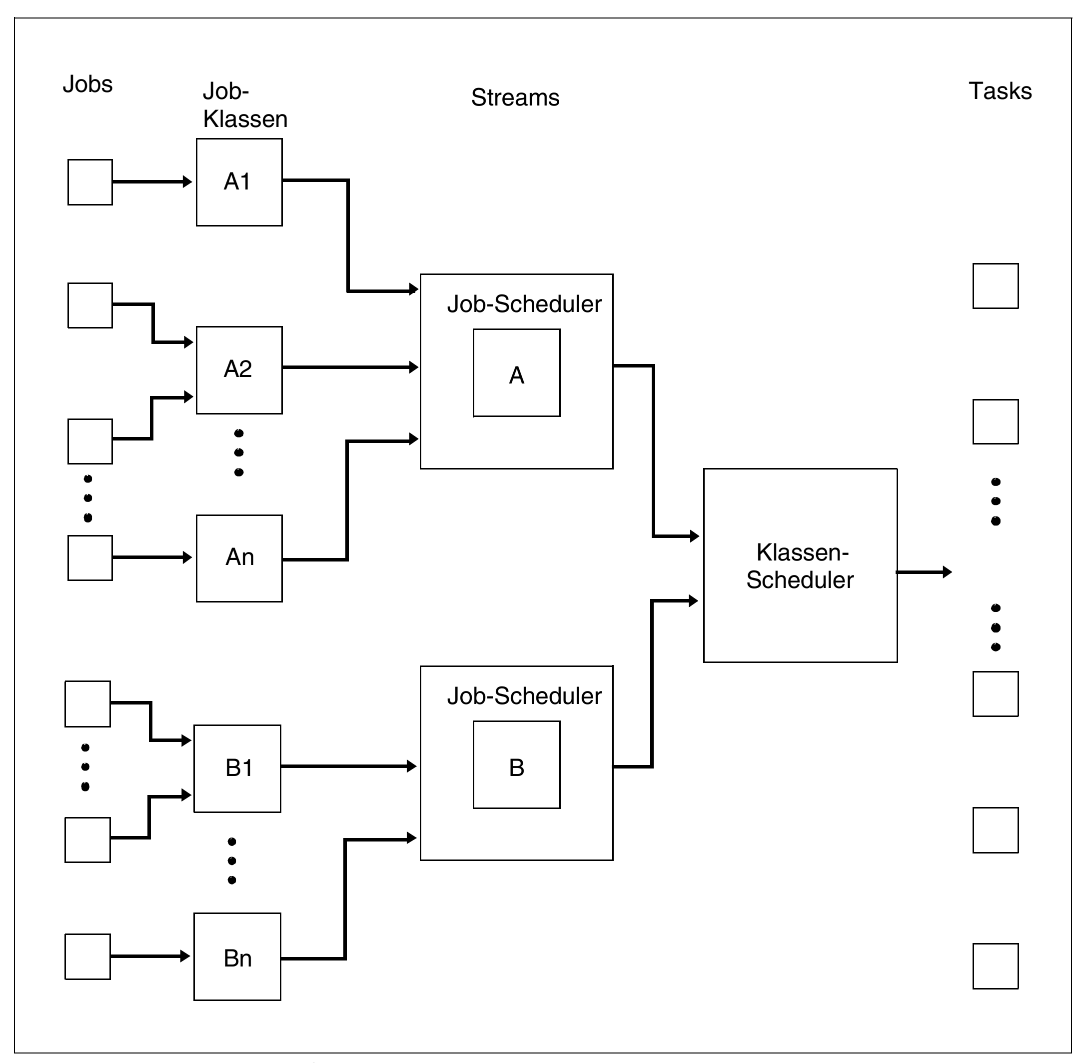



Bild 25: Komponenten der Job-Steuerung

### Unterstützung des zentralen Kalenders durch das JMS

Die Startzeitpunkte von Batch-Jobs können über symbolische Daten aus dem zentralen Kalender von BS2000/OSD angegeben werden. Dazu dient ein weiterer neuer Typ von Batch-Job: der Kalender-Job. Somit sind folgende drei Typen definiert:

der Termin-Job — ist ein einmalig ablaufender Batch-Job, dessen Startzeitpunkt von einer Zeitangabe des Benutzers abhängig ist.

der Repeat-Job — ist ein Wiederholungs-Job, dessen Wiederholungszyklus von einer Benutzerangabe (Repeat-Attribut) abhängig ist.

der Kalender-Job — ist entsprechend seiner Eigenschaft ein Termin-Job mit dem Start-Attribut *AT(...), der jedoch nach Vorgabe eines Kalenders wiederholt werden kann.
Ein Kalender-Job ist somit auch ein zeitlich durch die Angaben im Kalender (oder im Operanden LIMIT der Kommandos ENTER-JOB und MODIFY-JOB) begrenzter Wiederholungs-Job.

Der erste Ablauf und die folgenden Wiederholungen eines Kalender-Jobs werden vom System gleich behandelt. Das System unterscheidet nicht – wie bei Repeat-Jobs – in „Current“ und „Next Image“ eines Kalender-Jobs.
Alle Abläufe eines Kalender-Jobs haben die gleiche TSN. Damit ist eine bessere Identifizierung und Überwachung des Kalender-Jobs möglich. Die einzelnen Abläufe werden jeweils bis zum Erreichen des Startzeitpunkts vom Job-Scheduler verwaltet: Am Ende der Terminierung des Vorgängers wird der Nachfolger eingerichtet (gleiche TSN) und dem Scheduler mit dem Start-Attribut *AT(...) und der nächsten Startzeit des symbolischen Datums aus dem Kalender übergeben.

Die Möglichkeiten zur Angabe des Startzeitpunkts richten sich nach den Vorgaben des zentralen Kalenders (z.B. ist der Wiederholungszyklus nur auf Tage und nicht auf Stunden bezogen). Siehe dazu das Handbuch „CALENDAR“ [7].

Im Gegensatz zu Repeat-Jobs können Kalender-Jobs mit Hilfe von Monitor-Jobvariablen (MONJVs) überwacht werden. Die MONJV bleibt für die gesamte Lebensdauer des Kalender-Jobs geöffnet. Zusammen mit der gleichbleibenden TSN ist somit jederzeit der Zugriff auf den Job über die MONJV gewährleistet.

Ausgefallene Wiederholungen eines Kalender-Jobs werden – entsprechend den einfachen Termin-Jobs – mit dem abgelaufenen Startdatum nachgeholt. Anschließend wird aus dem Kalender der nächste Startzeitpunkt ermittelt.

Mit Hilfe des Operanden LIMIT der Kommandos ENTER-JOB und MODIFY-JOB ist eine Begrenzung der Lebensdauer eines Kalender-Jobs möglich. Limit-Überschreitungen die sich z.B. durch ausgefallene Wiederholungen oder Verzögerungen im Scheduler ergeben, werden jedoch zugelassen.
Die Lebensdauer kann ebenfalls durch das Setzen eines Ablaufzählers begrenzt werden. Ist das Limit erreicht, wird der gesamte Wiederholungs-Job beendet.

| **Kommando** | **Bedeutung** |
|---|---|
| CANCEL-JOB [1] | Benutzer-Job abbrechen |
| CHANGE-TASK-PRIORITY [1] | Job- oder Task-Priorität eines Dialog-, Batch- oder Druck-Jobs ändern |
| ENTER-JOB | Kommandofolge, die in einer ENTER-Datei als Batch-Job gespeichert ist, unter Angabe von Job-Klasse und Job-Priorität starten |
| ENTER-PROCEDURE | Kommandofolge starten, die in einer Prozedur als Batch-Job gespeichert ist |
| EXIT-JOB | die Task beenden |
| FORCE-JOB-CANCEL | Benutzer-Job abbrechen |
| HOLD-JOB | Benutzer-Job in den Wartezustand versetzen |
| HOLD-JOB-CLASS | Job-Klasse in den Wartezustand versetzen |
| HOLD-JOB-STREAM | Job-Stream und Job-Scheduler in den Wartezustand versetzen |
| MODIFY-JOB [1] | Eigenschaften eines Benutzer-Jobs ändern |
| MODIFY-JOB-CLASS | Grenzen und Gewichtung von Job-Klassen ändern |
| MODIFY-JOB-OPTIONS | Die mit ENTER-JOB, SET-LOGON-PARAMETERS und MODIFY-JOB-OPTIONS eingestellten Werte ändern |
| MODIFY-JOB-STREAM | Ablaufpriorität der Stream-Task und stream-spezifische Parameter ändern |
| MODIFY-JOB-SWITCHES | Einstellung von Auftragsschaltern ändern |
| MOVE-JOBS | Job-Beschreibungen exportieren bzw. importieren |
| RESUME-JOB | Wartezustand eines Benutzer-Jobs aufheben |
| RESUME-JOB-CLASS | Wartezustand einer Job-Klasse aufheben |
| RESUME-JOB-STREAM | Wartezustand von Job-Stream und Job-Scheduler aufheben |
| SET-LOGON-PARAMETERS | Einleiten eines Dialog- oder Batch-Jobs unter Angabe von Job-Klasse bzw. Job-Priorität |
| SHOW-JOB-CLASS | Informationen über Job-Klassen-Attribute;<br>unterstützt die Ausgabe in S-Variable |
| SHOW-JOB-OPTIONS | Informationen über die mit ENTER-JOB, SET-LOGON-PARAMETERS und MODIFY-JOB-OPTIONS eingestellten Werte anfordern |
| SHOW-JOB-STATUS [1] | Informationen über einen Benutzer-Job anfordern |
| SHOW-JOB-STREAM | Beschreibung aller Job-Streams anfordern |
| SHOW-SYSTEM-STATUS | Informationen über Job-Klassen und Job-Streams anfordern |
| SHOW-USER-ATTRIBUTES | Berechtigung zu den verschiedenen Job-Klassen abfragen |

Tabelle 29: Schnittstellenübersicht zur Job-Steuerung (Teil 1 von 2)

| **Kommando** | **Bedeutung** |
|---|---|
| SHOW-USER-STATUS [1] | Informationen über Gruppen von Benutzer-Jobs anfordern |
| START-JMP | Programm zur Rekonstruktion von ENTER-JOB-Kommandos starten |
| START-JOB-STREAM | Job-Stream und Job-Scheduler starten |
| STOP-JOB-STREAM | Job-Stream und Job-Scheduler beenden |
| **Makro** | **Bedeutung** |
| DJINF | DSECT oder Datenbereich für den Makro JINF erstellen |
| ENTER | ENTER-Job abarbeiten |
| JINF | Job-Daten anfordern |
| JMGDJP | DSECT oder Datenbereich für den Makro JMGJPAR erstellen |
| JMGJPAR | Job-Parameter anfordern |
| JOBINFO | Job-Daten ausgewählter Jobs anfordern |
| JSATTCH | Job-Scheduler mit dem JMS verbinden |
| JSDETCH | Job-Scheduler vom JMS lösen |
| JSEXPCT | JSS-Ereignisse anfordern |
| JSINFO | Stream-Parameter abfragen |
| JSRUNJB | Job zum Start übergeben |
| JSWAKE | Zeitereignis für Job-Scheduler initiieren |
| LGOFF | Job beenden |
| SWITCH | Benutzer- und Auftragsschalter abfragen und verändern |
| TMODE | Auftrags-Attribute abfragen |

Tabelle 29: Schnittstellenübersicht zur Job-Steuerung (Teil 2 von 2)

[1] Durch diese Kommandos können auch Batch-Jobs bearbeitet werden, die von der aufrufenden Benutzerkennung erzeugt wurden, aber unter einer fremden Benutzerkennung ablaufen

### 11.1.1 Konzept der Job-Klassen

Die Verwendung von Job-Klassen ermöglicht der Systembetreuung eine Klassifizierung von Benutzer-Jobs.

Das Konzept der Job-Klassen trägt verschiedenen Anforderungen Rechnung.

1. Über die Job-Klassen und deren Zuordnung zu den Job-Schedulern kann ein optimaler Auftragsmix erzeugt werden (z.B. viele kurzlaufende Jobs, wenige langlaufende Jobs). Letztendlich trägt dies zu einer **ausgewogenen Systemauslastung** bei. Außerdem besteht die Möglichkeit, mit Hilfe der Zugriffsrechte zu den Job-Klassen, Privilegien der Job-Steuerung unter den Benutzern zu verteilen.
2. Eine **zusätzliche Zugangskontrolle** zum System kann erreicht werden, indem die Systembetreuung den Benutzern Nutzungsrechte für bestimmte Job-Klassen gibt. Gibt der Benutzer in den Kommandos SET-LOGON-PARAMETERS bzw. ENTER-JOB keine Job-Klasse an, so wird sein Job in einer Standard-Job-Klasse geführt.
   Die Systembetreuung hat die Möglichkeit, Job-Klassen zu definieren, zu modifizieren oder zu löschen. Dazu steht das Dienstprogramm JMU zur Verfügung, das im Handbuch „Dienstprogramme“ [15] beschrieben ist.
3. Da sich Jobs durch die Benutzeranforderungen unterscheiden, müssen sie auch von den Job-Schedulern unterschiedlich behandelt werden.

Die Beschreibung einer Job-Klasse basiert auf einer Vielzahl von Angaben. Auf Grund vielschichtiger Möglichkeiten, Job-Eigenschaften zu kombinieren, ist es theoretisch denkbar, für jede Job-Merkmal-Kombination eine eigene Job-Klasse zu definieren.
Eine solche Vorgehensweise geht allerdings stark zu Lasten der Transparenz.

Die Systembetreuung sollte demnach die Job-Klassen nach den Kriterien definieren, die für den täglichen Produktionsbetrieb wichtig sind.

Mit dem Dienstprogramm JMU (Anweisung DEFINE-JOB-CLASS) legt die Systembetreuung u.a. folgende Eigenschaften und Vereinbarungen für eine Job-Klasse fest:

- Job-Klassenname
- Zuständiger Stream oder Default-Stream
- Dringlichkeit (Gewichtung) der Job-Klasse
- maximale Anzahl von Jobs, die in der Job-Klasse gleichzeitig ablaufen können
- Anzahl der Jobs, die idealerweise in der Job-Klasse laufen sollen
- Auftragstyp
- Job- und Task-Scheduling-Priorität
- Erlaubnis für das Starten von Repeat-Jobs
- Wiederholungsrhythmus für Repeat-Jobs
- maximal zu verbrauchende CPU-Zeit
- Start-Attribute

Während die **Job-Scheduling-Priorität** das Starten des Jobs beeinflusst, bezieht sich die **Task-Scheduling-Priorität** auf den Ablauf des gestarteten Jobs (= Ausführungspriorität).

Als mögliche Werte gelten für die Job-Steuerung die Prioritäten 1 bis 9 und für die Task-Steuerung die Prioritäten 30 bis 255.

Je niedriger der angegebene Wert ist, desto höher ist die Priorität.

Der Begriff **Auftragstyp** trifft eine Unterscheidung zwischen Batch- und Dialog-Jobs. Für beide Auftragstypen lassen sich Job-Klassen definieren, wobei allerdings zu beachten ist, dass die Dialog-Job-Klassen keinem Job-Scheduling unterliegen.
Hier wird lediglich auf die Einhaltung der Klassen-Grenzen geachtet, d.h. auf die Zahl der in der Job-Klasse geführten Dialog-Jobs und die Zugangsberechtigung zu dieser Job-Klasse.

*Beispiel 1*

Einteilung der Job-Klassen nach der zu verbrauchenden CPU-Zeit:

– Job-Klasse JCSHORT
  für Jobs, die nicht mehr als 5 CPU-Sekunden verbrauchen werden
– Job-Klasse JCNORMAL
  für Jobs, die nicht mehr als 500 CPU-Sekunden verbrauchen werden
– Job-Klasse JCLONG
  für Jobs, die mehr als 500 CPU-Sekunden verbrauchen werden

*Beispiel 2*

Einteilung der Job-Klassen nach dem Start-Zeitpunkt:

– Job-Klasse JCEXPRES
  für Jobs, die mit dem Start-Attribut IMMEDIATE ausgerüstet sind
– Job-Klasse JCNORMAL
  für Jobs, die keinen besonderen Zeitpunkt zum Start aufweisen
– Job-Klasse JCTERMIN
  für Jobs, die zu einem bestimmten Zeitpunkt (Datum/Uhrzeit) gestartet werden sollen (Termin-Jobs)

Neben den Job-Klassen für Benutzer-Jobs existiert die vordefinierte System-Job-Klasse mit Namen $SYSJC für System-Jobs. Diese System-Job-Klasse sollte den Benutzern nicht zur Verfügung stehen, da $SYSJC alle Auftragstypen zulässt und keinerlei Begrenzung hinsichtlich der Klassenattribute enthält.

Es bestehen grundsätzlich zwei Möglichkeiten, Job-Klassen zu definieren, zu modifizieren oder zu löschen:

1. Die **statische** Definition ist in der Datei $TSOS.SJMSFILE hinterlegt. Diese Datei ist die Basis für jeden Systemlauf. Sie wird mit dem Dienstprogramm JMU erzeugt und verwaltet.

   Folgende JMU-Anweisungen stehen zur Verfügung:

   ```
   DEFINE-JOB-CLASS
   MODIFY-JOB-CLASS
   DELETE-JOB-CLASS
   GRANT-JOB-CLASS-ACCESS
   SET-JOB-CLASS-DEFAULT
   SET-POSIX-JOB-CLASS-DEFAULT
   ```

   Will die Systembetreuung die Attribute einer Job-Klasse modifizieren oder die Zuordnung einer Benutzerkennung zu einer Job-Klasse in der Datei $TSOS.SJMSFILE ändern, so wirken sich diese Änderungen erst im nächsten Systemlauf aus.

2. Die **dynamische** Definition bezieht sich nur auf den aktuellen Systemlauf. Sie wird ebenfalls mit dem Dienstprogramm JMU verwaltet, wobei der Bearbeitungsmodus `SET-MODIFICATION-MODE=*SYSTEM` eingestellt sein muss. Es stehen die selben JMU-Anweisungen wie oben zur Verfügung. Bei `SET-MODIFICATION-MODE=*ALL` werden die Änderungen auch in die Datei $TSOS.SJMSFILE übernommen.

   Darüber hinaus gibt es Eingriffsmöglichkeiten über die Kommandoschnittstelle.
   Mit folgenden Kommandos reagiert die Systembetreuung kurzfristig und schnell auf Überlastsituationen, ohne die Systemdatei $TSOS.SJMSFILE ändern zu müssen:

   ```
   HOLD-JOB-CLASS
   MODIFY-JOB-CLASS
   RESUME-JOB-CLASS
   ```

### 11.1.2 Job-Streams, Job- und Klassen-Scheduler

Die Job-Klassen werden unterschiedlichen Job-Schedulern zugeordnet (mit dem Dienstprogramm JMU, siehe Handbuch „Dienstprogramme“ [15]).

Jeder Job-Scheduler läuft in einem eigenen Job ab, genannt „Job-Stream“.

Gemäß den vereinbarten Job-Scheduling-Strategien entscheiden die Job-Scheduler, welcher der Jobs zum Start freigegeben wird.

Job-Streams – und damit implizit die Job-Scheduler – steuern die Auswahl der Benutzer-Jobs, indem jedem Job-Stream ein passender Job-Scheduling-Algorithmus zur Verfügung steht.

*Beispiel*

In der Regel hat jedes Data Center verschiedene Produktionsschwerpunkte:

1. Die Programmierung benötigt für ihre Jobs zum Übersetzen, Binden und Laden von Programmen keine speziellen Betriebsmittel.
2. Die Jobs der Fachabteilungen sind mit wenigen Ausnahmen langlaufende Jobs mit hoher Priorität und hohen Betriebsmittelanforderungen.
3. Das Data Center erstellt für administrative Zwecke eine Reihe von Jobs, die zu bestimmten Zeiten ablaufen und wiederholt werden müssen.

Es bietet sich an, für diese drei Produktionsgebiete drei Job-Streams (Job-Scheduler) mit den folgenden Job-Scheduling-Strategien zu installieren:

zu 1) Für die Jobs der Programmierung kommt ein auf dem FIFO-Prinzip basierender Job-Scheduler in Frage (FIFO = First In First Out).
Jobs zum Übersetzen, Binden von Programmen etc. können z.B. in Job-Klassen zusammengefasst werden, die einem Job-Scheduler zugeordnet sind, der nur nachts aktiv ist.
Damit kann den unterschiedlichen Belastungen, denen ein Data Center tagsüber ausgesetzt ist, Rechnung getragen werden.

zu 2) Für die Jobs der Fachabteilungen eignet sich ein auf Priorität und Betriebsmittelverbrauch basierender Job-Scheduler.

zu 3) Für die Jobs des Data Centers bietet sich ein Scheduling-Algorithmus an, der die Termin-Jobs berücksichtigt.

Die Systembetreuung bestimmt mit der JMU-Anweisung DEFINE-JOB-STREAM die Kriterien, die dem Job-Stream zu Grunde liegen sollen und gemäß denen der Job-Scheduler die Jobs dann steuert. So kann z.B. die Auftragspriorität oder die benötigte CPU-Zeit als Kriterium über den Operanden S-PAR angegeben werden.

Neben dem Scheduling-Algorithmus gibt es noch weitere Eigenschaften, die einen Job-Stream kennzeichnen:

- Ein Job-Stream muss nicht immer aktiv sein. Die Systembetreuung entscheidet, ob der Stream z.B. bereits nach dem Laden von BS2000/OSD oder erst in der Nacht aktiv sein soll. Jobs, die für einen inaktiven Stream eingelesen wurden, werden solange in einer speziellen Warteschlange (TYPE1/DO) gesammelt, bis der Stream aktiv ist und der Job-Scheduler die Verwaltung übernimmt.
- Bei der Definition des Job-Streams wird für den Stream eine Task-Scheduling-Priorität vereinbart, mit der der Stream ab dem Startzeitpunkt geführt wird. Da jeder Job-Stream in einem eigenen Job läuft, ist auf diese Weise die Rangfolge eindeutig. Mit dem Kommando MODIFY-JOB-STREAM kann diese Rangfolge verändert werden.
- Ein Job-Stream verwaltet nur Batch-Jobs. Dialog-Jobs unterliegen keinem Job-Scheduling und werden gestartet, wenn der Benutzer das Recht hat, in der jeweiligen Job-Klasse Dialog-Jobs zu starten und wenn die Job-Klasse ihr festgelegtes Limit noch nicht erreicht hat.

Zur Definition der Job-Streams und damit implizit der Job-Scheduler stehen zwei Möglichkeiten zur Verfügung:

1. Die **statische** Definition, die in der Datei $TSOS.SJMSFILE hinterlegt ist, ist die Basis für jeden Systemlauf. Sie wird mit dem Dienstprogramm JMU erzeugt und verwaltet.

   Die dazu notwendigen JMU-Anweisungen lauten:

   ```
   DEFINE-JOB-STREAM
   DELETE-JOB-STREAM
   MODIFY-JOB-STREAM
   ```

   Änderungen in dieser Datei wirken sich erst beim nächsten Systemlauf aus.

2. Die **dynamische** Definition bezieht sich nur auf den aktuellen Systemlauf. Sie wird ebenfalls mit dem Dienstprogramm JMU verwaltet, wobei der Bearbeitungsmodus `SET-MODIFICATION-MODE=*SYSTEM` eingestellt sein muss. Es stehen die selben JMU-Anweisungen wie oben zur Verfügung. Bei `SET-MODIFICATION-MODE=*ALL` werden die Änderungen auch in die Datei $TSOS.SJMSFILE übernommen.

   Darüber hinaus gibt es Eingriffsmöglichkeiten über die Kommandoschnittstelle.

   Mit folgenden Kommandos reagiert die Systembetreuung kurzfristig und schnell auf Überlastsituationen, ohne die in der Datei $TSOS.SJMSFILE hinterlegten Definitionen ändern zu müssen:

   ```
   HOLD-JOB-STREAM
   MODIFY-JOB-STREAM
   RESUME-JOB-STREAM
   START-JOB-STREAM
   STOP-JOB-STREAM
   ```

Der Name der ENTER-Datei, die den Job-Stream definiert, ist in der Datei $TSOS.SJMSFILE hinterlegt. Der Name ist prinzipiell frei wählbar, aus Konsistenzgründen wird SYSENT.JOBSCHED.nnn empfohlen („nnn“ ist das Kennzeichen der Betriebssystem-version).

Der Job-Scheduler wird aus dieser Datei mit START-EXECUTABLE-PROGRAM aufgerufen, womit sein Name festgelegt wird.
Auch der Name des Job-Schedulers ist somit prinzipiell frei wählbar; aus Gründen der Konsistenz, für die die Systembetreuung verantwortlich ist, wird der Name SYSPRG.JOBSCHED.nnn empfohlen („nnn“ ist das Kennzeichen der Betriebssystemversion).

Für den Stream-ENTER werden vereinbarte Logon- bzw. Logoff-Prozeduren nicht ausgeführt (zu Logon-/Logoff-Prozeduren siehe Handbuch „Dialogschnittstelle SDF“ [45]).

Von ihrer Bedeutung und Funktion sind zu unterscheiden:

- der System-Job-Scheduler oder Emergency-Job-Scheduler $SYSJS
- die Standard-Job-Scheduler
- der Klassen-Scheduler

### 11.1.2.1 System-Job-Scheduler

Im Gegensatz zum Standard-Scheduler ist der System- oder Emergency-Job-Scheduler Bestandteil des mit dem Betriebssystem gekoppelten Job-Streams $SYSJS.

Er läuft im Funktionszustand TPR ab und verwaltet nur Jobs, die der Systemklasse $SYSJC zugeordnet sind. Die Scheduling-Strategie, auf dem LIFO-Prinzip (Last In First Out) basierend, ist nicht beeinflussbar, das heißt, der START-Parameter wird ignoriert.

Dies ist vor allem dann von Vorteil, wenn bei der Inbetriebnahme der Job-Steuerung Fehler auftreten und wenn Fehler beim Eröffnen oder beim Zugriff auf die Dateien $TSOS.SJMSFILE oder $TSOS.SJMSFILE.WORK angezeigt werden.

Es besteht dann immer die Möglichkeit, die Dateien, entsprechend dem Fehlerfall, zu rekonstruieren und die entsprechenden Jobs zu starten.

Mit folgenden Kommandos können auch die Systemklasse $SYSJC und der System-Job-Scheduler $SYSJS bearbeitet werden:

```
HOLD-JOB-STREAM
HOLD-JOB-CLASS
RESUME-JOB-STREAM
RESUME-JOB-CLASS
SHOW-JOB-STREAM
SHOW-JOB-CLASS
```

Die Definitionen von $SYSJS und $SYSJC sind konstant und werden vom Dienstprogramm JMU nicht angesprochen.

Der System-Job-Scheduler lässt unabhängig vom Standard-Job-Scheduler während des gesamten Systemlaufs das Starten von Jobsn unter der Benutzerkennung TSOS zu. Wenn die Systembetreuung für die Benutzer keine Job-Klassen definiert hat, ist die System-Job-Klasse $SYSJC die Standard-Job-Klasse für die Benutzer, und damit verbunden werden alle Benutzer-Jobs vom System-Job-Scheduler $SYSJS bearbeitet.

### 11.1.2.2 Standard-Job-Scheduler

Der Standard-Job-Scheduler entscheidet, welcher der Jobs der ihm zugeordneten Job-Klassen zum Start freigegeben wird.

Dabei bedient sich der Standard-Job-Scheduler einer festgelegten Job-Scheduling-Strategie, die bei der statischen Definition mit dem Dienstprogramm JMU festgelegt und mit dem Kommando MODIFY-JOB-STREAM dynamisch verändert werden kann.
Gleichzeitig können maximal 16 Standard-Job-Scheduler mit unterschiedlichen Scheduling-Strategien installiert sein.
Ein Standard-Job-Scheduler läuft in einem eigenen Job (Job-Stream) ab. Es sind Benutzerprogramme im Funktionszustand TU unter der Kennung der Systemverwaltung.

Sollten die Auftragssteuerungs-Möglichkeiten des Standard-Job-Schedulers für die Belange des Data Centers nicht ausreichen, lässt sich ein selbst entwickelter Job-Scheduler zum Einsatz bringen (siehe hierzu Abschnitt „Job-Steuerung im Data Center“ auf Seite 470).

In beiden Fällen wird der Scheduler unter der Kennung der Systemverwaltung gestartet (automatisch oder mit dem Kommando START-JOB-STREAM, je nachdem was für den Operanden START in der Streamdefinition des Schedulers im SJMSFILE angegeben wurde).

#### Job-Scheduling-Parameter

Bei der Stream-Definition mit der JMU-Anweisung DEFINE-JOB-STREAM werden mit dem Operanden STREAM-PARAMETER (kurz S-PAR) scheduler-spezifische Parameter definiert.

Dieser Parameter bleibt von BS2000/OSD unberücksichtigt; er wird lediglich vom Job-Scheduler ausgewertet und liefert die Basis für die Berechnung des Scheduling-Algorithmus.

Für den Standard-Job-Scheduler werden mit dem S-PAR-Operanden folgende Parameter vereinbart:

```
S-PAR =  'JOB-PRIORITY = YES / NO
         ,CPU-TIME = NO / YES
         ,WAIT-TIME = NO / YES
         ,JOB-QUOTA = 1 / no<256
         ,LOGGING = YES / NO
         ,CATID-LIST = (catid1,...)
         ,CAT-TIME = min'
```

*Hinweise*

1. Die Definitionsvarianten der ersten drei Parameter entsprechen den in Tabelle 30 auf Seite 462 aufgeführten Scheduling-Strategien.

2. Aus der Reihe der Jobs werden die Jobs ausgewählt, für die bei der Berechnung des Scheduling-Algorithmus die niedrigste Rangfolge ermittelt wird.
Die Anzahl der ausgewählten Jobs wird vom Parameter JOB-QUOTA bestimmt.
Der Job-Scheduler versucht nun diese Jobs gemäß ihrer Reihenfolge in der Warteschlange zu starten. Nach erfolgtem Start werden wiederum Jobs aus der Reihe der Jobs ausgewählt. Dies wiederholt sich so lange, bis alle Jobs gestartet sind oder das Job-Klassen-Limit erreicht ist.

3. Der LOGGING-Parameter steuert die Protokollausgabe des Standard-Job-Schedulers. Außerdem legt der LOGGING-Parameter fest, ob eventuell vom Job-Scheduler erzeugte Meldungen an der Bedienstation ausgegeben werden sollen.

4. Zu diesen Meldungen zählen:

   – `JMS0302`: Fehler beim Ausführen eines ASSIGN-SYSFILE-Kommandos
   – `JMS0303`: Protokollausgabe wegen Speicherplatzmangel abgebrochen
   – `JMS0304`: Nicht behebbarer Fehler
   – `JMS0305`: Ungültiger Stream-Parameter
   – `JMS0306`: Job-Stream wartet auf die Verfügbarkeit der angegebenen Pubsets
   – `JMS0307`: Job-Stream wird wegen fehlender Pubsets beendet
   – `JMS0308`: Job-Stream wartet auf Verfügbarkeit des Subsystems GET-TIME

5. Der Parameter CATID-LIST vereinbart die Pubsets, die vor der Inbetriebnahme des Job-Schedulers importiert sein müssen. Der Parameter wird nur zum Zeitpunkt des Stream-Starts berücksichtigt und verzögert das Starten des Job-Schedulers, bis sämtliche, bei CATID-LIST angegebenen Pubsets verfügbar sind. Fehlt der Parameter CATID-LIST, so startet der Job-Scheduler unabhängig davon, welche Pubsets importiert sind.
Zur Steuerung dieser Wartezeit dient der Parameter CAT-TIME.
Kann ein Pubset innerhalb der bei CAT-TIME angegebenen Zeit nicht importiert werden, wird der Job-Scheduler nicht gestartet.
Fehlt der Parameter CAT-TIME in der Stream-Definition, wartet das System beliebig lange auf das Importieren der benötigten Pubsets. Die Zeitangabe erfolgt beim Parameter CAT-TIME in Minuten.

6. Über die Kommandoschnittstelle kann die Systembetreuung die Einstellung des Operanden S-PAR dynamisch ändern (siehe Kommando MODIFY-JOB-STREAM im Handbuch „Kommandos“ [27]).

### Job-Scheduling-Algorithmus

Die Job-Scheduler übergeben ihre zum Start freigegebenen Jobs an den Klassen-Scheduler. Die Reihenfolge, innerhalb der die Jobs zum Start weitergegeben werden, wird mit folgender Formel errechnet:

$$M = \frac{S^a \times P^c \times R^d}{W^b + S^{(a \times b)}}$$

Dabei bedeutet:

M Rangfolge

S CPU-Zeit in Sekunden, die der Job angefordert hat

P Job-Scheduling-Priorität

W Wartezeit des Auftrags in Minuten nach Job-Annahme

R Zeit in Minuten, die beim Starten des Jobs noch verbleibt, bis der späteste Startzeitpunkt erreicht ist.
Voraussetzung ist die Verwendung des Start-Attributes START=WITHIN oder START=LATEST.

a,b,c Job-Scheduler-Parameter, die den Wert 0 oder 1 annehmen können, werden von der Systembetreuung bei der Stream-Beschreibung festgelegt. Dabei entspricht:

| | |
|---|---|
| CPU-TIME | der Variablen a |
| WAIT-TIME | der Variablen b |
| OB-PRIORITY | der Variablen c |

d d=1, wenn der Job vom Benutzer mit den Start-Attributen START=WITHIN oder START=LATEST bereitgestellt wird.
Ansonsten gilt: d=0.



Je kleiner der Wert für M ist, desto besser ist die Position des Jobs, dies gilt unabhängig von der Job-Klasse.

Wird für mehrere Jobs der gleiche Wert M ermittelt, dann wird die Reihenfolge der zum Start freigegebenen Jobs nach dem FIFO-Prinzip bestimmt.

Im Folgenden werden die einzelnen Scheduling-Verfahren erläutert, die sich aus den verschiedenen Kombinationen der Exponentenwerte ergeben. Die Wertzuweisung für die Exponenten a, b, c erfolgt bei der Stream-Definition durch den Operanden S-PAR.

Die Scheduling-Strategien sind in der Tabelle nur mit der jeweiligen Abkürzung angegeben. Die dazu gehörende Langform ist im Anschluss an die Tabelle vermerkt.

| **Scheduling-Strategie** | **CPU-TIME (a)** | **WAIT-TIME (b)** | **JOB-PRIORITY (c)** |
|---|---|---|---|
| (1) FIFO | NO | YES | NO |
| (2) HPF | NO | NO | YES |
| (3) HPA | NO | YES | YES |
| (4) SJF | YES | NO | NO |
| (5) SJP | YES | NO | YES |
| (6) HRN | YES | YES | NO |
| (7) HRP | YES | YES | YES |

Tabelle 30: Scheduling-Strategien

(1) *Auswahl nach Ankunftszeit (FIFO)*

Die Strategie empfiehlt sich, wenn die CPU-Zeitanforderung der freizugebenden Jobs nicht stark voneinander abweicht.
Ist dies nicht der Fall, dann führt die ausschließliche Verwendung dieser Strategie dazu, dass langlaufende Jobs bevorzugt werden.

Auswahl der Jobs, die mit der Option START=WITHIN bzw. START=LATEST beim ENTER-JOB-Kommando versehen wurden (FIFO und R (Wartezeit)):

Basierend auf dem FIFO-Auswahlprinzip wird der STREAM-Parameter WAIT-TIME=YES ausgewertet.
Sind die Jobs mit der Start-Option START=WITHIN in das System gekommen, dann werden die Jobs erst ab Beginn des Startintervalls in die Auswahlentscheidung einbezogen.
Für Jobs mit START=LATEST wächst die Wahrscheinlichkeit, gestartet zu werden, mit der Annäherung an die spätest gewünschte Startzeit.

FIFO kann bei WAIT-TIME=YES nicht garantiert werden, wenn ein Stream-Start läuft und ein TSN-Überlauf aufgetreten ist. Wurden zwei Jobs in derselben Minute gestartet und tritt gerade dann ein TSN-Überlauf auf, wird der Job mit der TSN 0AAA vor dem Job mit der letzten vergebbaren TSN (9999) gestartet.

(2) *Auswahl nach Priorität (HPF)*

Die Jobs werden gemäß ihrer extern zugewiesenen Job-Scheduling-Priorität ausgewählt. Da immer der Job zum Start freigegeben wird, der die beste Priorität nachweisen kann, ist die Bevorzugung privilegierter Jobs gewährleistet.

Auswahl der Jobs, die mit der Option START=IMMEDIATE bzw. START=AT beim ENTER-JOB-Kommando versehen wurden:
Diese Funktionen bewirken, dass der Scheduling-Algorithmus den Wert M (Rangfolge) für Jobs mit der Option START=IMMEDIATE automatisch auf 0 setzt. Für Jobs mit der Option START=AT wird M auf 0 gesetzt, wenn die angegebene Startzeit erreicht wurde. Jobs für die M=0 ermittelt wird, werden sofort zum Starten freigegeben und können von Jobs mit M>0 nicht überholt werden.

(3) *Auswahl nach Priorität und Alterung (HPA)*

Durch Einbeziehung der Wartezeit in die Auswahlstrategie wird erreicht, dass selbst bei permanentem Angebot hochpriorer Jobs auch solche mit niedriger Priorität zum Klassen-Scheduler weitergeleitet werden.

(4) *Auswahl nach Laufzeit (SJF)*

Diese Strategie bevorzugt kurzlaufende Jobs und verkürzt im Gegensatz zur FIFO-Methode die mittlere Wartezeit. Da Batch-Jobs bei Ablauf der angeforderten CPU-Zeit abgebrochen werden, kann die Strategie nicht unterlaufen werden. SJF verzögert den Start von langlaufenden Jobs und kann im Extremfall – nämlich bei genügend kurzlaufenden Jobs – dazu führen, dass langlaufende Jobs gar nicht zum Start freigegeben werden.

(5) *Auswahl nach CPU-Zeit und Priorität (SJP)*

Bei dieser Strategie werden die Methoden „Auswahl nach Laufzeit“ und „Auswahl nach Priorität“ kombiniert.
Bei Jobs mit gleicher CPU-Zeit-Anforderung ist die Priorität maßgebend; bei Jobs mit gleicher Job-Scheduling-Priorität entscheidet die angeforderte CPU-Zeit.

(6) *Auswahl nach Response Ratio (HRN)*

Response Ratio ist das Verhältnis (Wartezeit + CPU-Zeit) / CPU-Zeit und kombiniert die beiden Verfahren HPA und SJF. Obwohl wiederum die kurzlaufenden Jobs bevorzugt werden, wird jedoch die Wartezeit der Jobs berücksichtigt.

(7) *Auswahl nach Durchsatz und Priorität (HRP)*

Im Vordergrund steht die Bevorzugung von kurzlaufenden Jobs. Die Rangfolge der Jobs ist festgelegt über die jeweilige Job-Scheduling-Priorität.

### Protokoll des Standard-Job-Schedulers

Der Standard-Job-Scheduler protokolliert bestimmte Ereignisse der Stream-Tasks in die Protokolldatei SYS.SCHEDLOG.<yyyy.mm.dd>.<hh.mm.ss>.<streamname>, wenn in der Stream-Definition die Protokollierung eingeschaltet ist
(in der JMU-Anweisung DEFINE-JOB-STREAM oder im Kommando MODIFY-JOB-STREAM, Operand S-PAR='...,LOGGING=YES').
Das Protokoll gibt einen Überblick über die Aktivitäten des Schedulers während eines Systemlaufs. Darüber hinaus dient es als Unterlage bei einer evtl. erforderlichen Fehleranalyse.

Ausschnitt aus einer Protokolldatei (die grau unterlegten Nummern sind nicht Teil der Protokolldatei, sondern dienen der Zuordnung zum erläuternden Text):

```
FTS BS2000   JOB SCHEDULER JSSTD1 VERSION <version> <date> <time>    PAGE 1
**             STREAM STARTED
** JSSTD1      JSATTCH (1)
**             EVENT          10   (JS_HOLD) (16)
**             EVENT          11   (JS_RELEASE) (17)
**             EVENT           9   (JC_AVAILABLE) (15)   JCLASS=JCDSTD
**             JSWAKE (4)                                 WAKE-TIME=2147483647
**             EVENT           9   (JC_AVAILABLE) (15)   JCLASS=JCDSTD
**             EVENT           9   (JC_AVAILABLE) (15)   JCLASS=JCDSTD
**             EVENT           9   (JC_AVAILABLE) (15)   JCLASS=JCDSTD
**             EVENT               (J_INTRODUCTION) (6)  TSN=0FSM
**             JSRUNJB (3)                                TSN=0FSM, JCLASS=JCBSTD,
                                                          WAIT-TIME=0, JOB-PRIO=9,
                                                          CPU-TIME=32000,
                                                          MERIT=45DBBA000000000000000001
**             EVENT           1   (J_TERMINATION) (7)   TSN=0FSM
**             EVENT           9   (JC_AVAILABLE) (15)   JCLASS=JCBSTD
**             EVENT               (J_INTRODUCTION) (6)  TSN=0FSN
**             JSRUNJB (3)                                TSN=0FSN, JCLASS=JCBSTD,
                                                          WAIT-TIME=0, JOB-PRIO=9,
                                                          CPU-TIME=32000,
                                                          MERIT=45DBBA000000000000000002
**             EVENT           9   (JC_AVAILABLE) (15)   JCLASS=JCDSTD
**             EVENT           1   (J_TERMINATION) (7)   TSN=0FSN
**             EVENT           9   (JC_AVAILABLE) (15)   JCLASS=JCBSTD
**             EVENT               (J_INTRODUCTION) (6)  TSN=0FSQ
**             JSRUNJB (3)                                TSN=0FSQ, JCLASS=JCBSTD,
                                                          WAIT-TIME=0, JOB-PRIO=9,
                                                          CPU-TIME=32000,
                                                          MERIT=45DBBA000000000000000003
**             EVENT               (J_INTRODUCTION) (6)  TSN=0FSR
```

```
**              JSRUNJB (3)                                          TSN=0FSR, JCLASS=JCBSTD,
                                                                     WAIT-TIME=0, JOB-PRIO=9,
                                                                     CPU-TIME=32000,
                                                                     MERIT=45DBBA000000000000000004
**              EVENT                 (J_INTRODUCTION) (6)           TSN=0FSS
**              JSWAKE (4)                                           WAKE-TIME=0
** JOB_HOLD     JOB HELD                                             TSN=0FSS
**              EVENT           2     (J_HOLD) (8)                   TSN=0FSS
**              JSWAKE (4)                                           WAKE-TIME=10111560
**              EVENT           4     (J_CANCEL) (10)                TSN=0FSS
**              JSWAKE (4)                                           WAKE-TIME=2147483647
**              EVENT                 (J_INTRODUCTION) (6)           TSN=0FSS
**              JSWAKE (4)                                           WAKE-TIME=0
**              EVENT          16     (JS_TIMER) (21)                DATE=<date>, TIME=<time>
**              JSWAKE (4)                                           WAKE-TIME=10111500
**              EVENT           3     (J_RELEASE) (9)                TSN=0FSS
**              EVENT           1     (J_TERMINATION) (7)            TSN=0FSQ
**              EVENT           9     (JC_AVAILABLE) (15)            JCLASS=JCBSTD
**              EVENT           1     (J_TERMINATION) (7)            TSN=0FSR
**              EVENT           9     (JC_AVAILABLE) (15)            JCLASS=JCBSTD
**              EVENT           4     (J_CANCEL) (10)                TSN=0FSS
**              JSWAKE (4)                                           WAKE-TIME=2147483647
**              EVENT          13     (JS_CLOSE_IMMEDIATE) (19)
**              STREAM CLOSED
** JSSTD1       JSDETCH (2)
```

Erläuterung der Funktionen des Job-Schedulers (über die Protokollierung sichtbare Schritte sind grau unterlegt):

(1) *JSATTCH*
Der Job-Stream informiert mit einer Meldung über seine abgeschlossene Inbetriebnahme.

(2) *JSDETCH*
Der angegebene Job-Stream befindet sich in der Beendigungsphase. Die Job-Steuerung unterstützt den Job-Scheduler nicht mehr. Darüber hinaus wird auch die Task, in der der Job-Stream realisiert ist, beendet.

(3) *JSRUNJB*
Der Job-Scheduler fordert den Klassen-Scheduler auf, den angegebenen Job zu starten.
Falls für die Job-Klasse das CLASS-LIMIT erreicht ist, erhält der Standard-Job-Scheduler eine entsprechende Rückinformation.

(4) *JSWAKE*
Der Job-Scheduler teilt der Job-Steuerung mit, wann er wieder aktiviert werden will.

(5) *JSEXPCT*
Die Job-Steuerung erhält die Aufforderung, dem Job-Scheduler das nächste vorliegende Ereignis zu übergeben. Dazu zählen dann (siehe Nummern 6 - 21):

(6) *J_INTRODUCTION*
Der Job-Scheduler wird aufgefordert, einen Job in die Menge der von ihm verwalteten Jobs aufzunehmen. Dabei werden dem Job-Scheduler folgende Informationen zur Verfügung gestellt:
Auftragsnummer, Name der Job-Klasse, Wiederholungsanzeige, CPU-Zeit, Job-Scheduling-Priorität, Ankunftszeit des Jobs, Start-Attribut, Job-Parameter.

(7) *J_TERMINATION*
Ein vom betreffenden Job-Scheduler verwalteter Job wird nach seiner Laufzeit beendet.

(8) *J_HOLD*
Ein Job wird angehalten und steht in der entsprechenden Warteschlange.

(9) *J_RELEASE*
Ein angehaltener Job wird wieder freigegeben. Der Job-Scheduler ist aufgefordert, den Job zu berücksichtigen.

(10) *J_CANCEL*
Ein vom Job-Scheduler noch nicht gestarteter Job wird abgebrochen.

(11) *J_EXPRESS*
Annahme eines Jobs, der sobald wie möglich gestartet werden soll. Der Job erhält die Rangfolge M=0 (siehe „Job-Scheduling-Algorithmus“ auf Seite 461).

(12) *J_RESCHEDULE*
Eine mit dem Kommando MODIFY-JOB vorgenommene Änderung der Job-Eigenschaften wird angezeigt.

(13) *JC_HOLD*
Die angegebene Job-Klasse wird in den Wartezustand versetzt. Der Job-Scheduler ist aufgefordert, keine Jobs der angehaltenen Job-Klasse weiter zu berücksichtigen.

(14) *JC_RELEASE*
Die angehaltene Job-Klasse und damit implizit alle Jobs dieser Job-Klasse werden wieder freigegeben.

(15) *JC_AVAILABLE*
Für die angegebene Job-Klasse ist der bei CLASS-LIMIT festgelegte Wert unterschritten worden.

(16) *JS_HOLD*
Der angegebene Job-Stream wird in den Wartezustand versetzt.

(17) *JS_RELEASE*
Der angehaltene Job-Stream wird wieder freigegeben. Er ist angewiesen, alle von ihm zu verwaltenden Jobs zu bearbeiten.

(18) *JS_CLOSE QUIET*
Anzeige, dass sich das Betriebssystem in der Beendigungsphase befindet. Dieses Ereignis hat die gleiche Wirkung wie JS_HOLD, d.h. es werden keine weiteren Jobs gestartet.

(19) *JS_CLOSE IMMEDIATE*
Nach Bearbeitung des Kommandos STOP-JOB-STREAM wird der Job-Scheduler sofort beendet.

(20) *JS_CHANGE*
Es werden Informationen über geänderte STREAM-PARAMETER ausgegeben.

(21) *JS_TIMER*
In Minutenabständen wird der Job-Scheduler angewiesen, die Start-Attribute seiner Jobs zu überprüfen.

### 11.1.2.3 Klassen-Scheduler

Der Klassen-Scheduler übernimmt im Rahmen der Job-Steuerung folgende Aufgaben:

*Starten der Batch-Jobs*

Wie das Bild 25 auf Seite 448 zeigt, übergeben sämtliche Job-Scheduler ihre zum Start freigegebenen Jobs an den Klassen-Scheduler.
In Abstimmung mit der Task-Steuerung wird ein vom Job-Scheduler freigegebener Job vom Klassen-Scheduler sofort gestartet.

*Überwachen der Job-Klassengrenzen*

Mit der JMU-Anweisung DEFINE-JOB-CLASS und dem Operanden CLASS-LIMIT wird für jede Job-Klasse eine Grenze festgelegt, bis zu der Jobs gestartet werden können.
Die Systembetreuung hat damit – bezogen auf alle Job-Klassen – einen Job-Mix vereinbart, für dessen Einhaltung der Klassen-Scheduler verantwortlich ist. Wenn dem Klassen-Scheduler ein Job übergeben wird, überprüft er, ob das CLASS-LIMIT erreicht ist. Ist dies der Fall, wird der Job zurückgewiesen (Ausnahme: ein Job mit Start-Attribut IMMEDIATE). Sobald der Grenzwert wieder unterschritten ist, z.B. durch Job-Ende oder durch Erhöhung des Limits per Kommando MODIFY-JOB-CLASS, wird der entsprechende, die Job-Klasse verwaltende Scheduler informiert und ein Job an den Klassen-Scheduler weitergegeben.

*Zwischenspeichern von Jobs bei Systemsättigung*

Bei Sättigung des Seitenwechselbereichs oder des Hauptspeichers nimmt der Klassen-Scheduler von den einzelnen Schedulern zwar noch Jobs an, speichert sie aber in klassenspezifischen Warteschlangen zwischen.
Die Einreihung der Jobs in diese Warteschlangen erfolgt nach dem FIFO-Prinzip. Eine Ausnahme bilden die Jobs, die mit dem Start-Attribut IMMEDIATE (sog. Express-Jobs) versehen sind. Diese Jobs stehen in den Warteschlangen an vorderster Stelle. Treffen mehrere Express-Jobs beim Klassen-Scheduler ein, dann steht der zuletzt eingetroffene Job als erster in der Warteschlange.

*Starten dieser Jobs nach Aufheben der Sättigung*

Nach Aufheben der Sättigung startet der Klassen-Scheduler die Jobs, indem er eine Job-Klasse, deren Klassenoptimum noch nicht erreicht ist, mit dem kleinsten Wert M nach folgendem Algorithmus auswählt:

$$M = \frac{C+1}{W}$$

Dabei bedeutet:

M Rangfolge

C Anzahl der momentan in der Job-Klasse laufenden Jobs

W Gewichtung (Dringlichkeit) der Job-Klasse

Der Klassen-Scheduler startet den Job, der als erster in der Warteschlange steht.

Nach jedem Job-Start und jeder Job-Beendigung muss der Klassen-Scheduler die Rangfolge erneut berechnen.

Bei Einsatz des Software-Produkts PCS ist der Klassen-Scheduler mit dem Performance-Controller verbunden.

### 11.1.3 Die Job-Steuerung während der Systemeinleitung

Für die Job-Steuerung werden in der Phase der Systemeinleitung nachstehende Funktionen ausgeführt:

- Aus der Datei SJMSFILE werden die Job-Stream- und Job-Klassen-Definitionen sowie die Klassenzugriffsrechte eingelesen.
- Von der Datei SJMSFILE (Typ ISAM) wird eine Kopie unter dem Namen SJMSFILE.WORK (Typ SAM) angelegt.
- System- und Klassen-Scheduler werden aktiviert.
- Auf der Bedienstation wird zusätzlich die Anzahl der aus der SJMSFILE bzw. SJMSFILE.WORK gelesenen Objekte (Streams, Klassen, Zugriffslisten usw.) ausgegeben.

Sollte beim Startup der Fall eintreten, dass die Datei SJMSFILE defekt oder nicht vorhanden ist, übernimmt die im vorherigen Systemlauf erzeugte Datei SJMSFILE.WORK deren Funktionen. Da die SJMSFILE neu erzeugt werden muss, ist es zu empfehlen, einen Batch-Job mit dem entsprechenden JMU-Lauf bereitzuhalten.

### 11.1.4 Job-Steuerung im Data Center

Die Job-Steuerung bietet der Systembetreuung darüber hinaus Möglichkeiten in Richtung eines Job-Schedulings im Data Center an:

1. Installieren eines eigenen Job-Schedulers

   Über die Assembler-Schnittstelle kann ein individueller Job-Scheduler erstellt werden, der als Benutzerprogramm unter der Benutzerkennung der Systemverwaltung abläuft. Die dazu erforderlichen Makros sind im Handbuch „Makroaufrufe an den Ablaufteil“ [30] beschrieben.
   Mit der JMU-Anweisung DEFINE-JOB-STREAM (Operand S-PAR) definiert die Systembetreuung zusätzliche Parameter, die in ihrer Syntax und Semantik frei wählbar sind. Sie werden nur von dem individuellen Job-Scheduler ausgewertet.

2. Benutzen einer System-Exit-Routine

   Mit der JMU-Anweisung DEFINE-JOB-CLASS (Operand JOB-PAR) definiert die Systembetreuung zusätzliche Parameter, die in ihrer Syntax und Semantik frei wählbar sind. Sie werden nur von der System-Exit-Routine (System Exit 32) ausgewertet.

Die Notwendigkeit, zusätzliche Parameter zu definieren, könnte sich für ein Data Center aus der Tatsache ergeben, dass der Standard-Job-Scheduler beispielsweise innerhalb der Job-Klassen keinen Unterschied macht zwischen:

| | | |
|---|---|---|
| **sehr wichtigen Jobs** | - | Klassenattribut VIMP |
| **wichtigen Jobs** | - | Klassenattribut IMP |
| **weniger wichtigen Jobs** | - | Klassenattribut NIMP |

Die Systembetreuung vereinbart demnach für die betroffenen Job-Klassen:

```
DEFINE-JOB-CLASS...,JOB-PAR='VIMP'/'IMP'/'NIMP'
```

### 11.1.5 Unterbrechungsfreie Zeitumstellung

Für die Winter-/Sommerzeit-Umstellung muss aus Sicht der Job-Steuerung das System nicht mehr heruntergefahren und wieder neu gestartet werden:

Die Job-Steuerung arbeitet intern mit der UTC-Zeit, die monoton aufsteigend ist und Zeitumstellungen nicht kennt. Extern verwendet die Job-Steuerung die für die Systeminstallation eingestellte Zeit, die sog. Systemzeit, die auf der lokalen Zeit (LT, gültige gesetzliche Zeit am Installationsort) basiert. Das bedeutet, dass der Benutzer immer mit der gesetzlichen Ortszeit arbeiten kann.

Voraussetzung für die richtige Zeitumrechnung durch die Job-Steuerung sind die korrekten System-Parameterdateien mit Angaben für die Zeitumstellung (siehe Seite 87). Zeitangaben werden also immer als gesetzlich gültige Ortszeit unabhängig von Zeitsprüngen interpretiert, z.B. Startzeitpunkte für Termin-Jobs.

Die konsequente Interpretation von Zeitangaben als gesetzliche Zeit hat auch ein geändertes Start-Verhalten von Repeat-Jobs mit dem Repeat-Attribut „DAILY“ bzw. „WEEKLY“ zur Folge: Bei der Akzeptierung solcher Repeat-Jobs wird der Startzeitpunkt für die tägliche oder wöchentliche Wiederholung bestimmt. Auch dieser Startzeitpunkt wird nun konsequent als gesetzliche Zeit interpretiert. So wird z.B. ein Job, der zur Sommerzeit täglich um 17 Uhr ablief, auch nach dem Wechsel auf Winterzeit täglich um 17 Uhr gestartet.

### 11.1.6 JMP: Rekonstruktion von Batch-Jobs

Das Programm JMP (Jobpool Management Program) rekonstruiert aus den Informationen, die das JMS über akzeptierte Batch-Jobs im Job-Pool (in der Systemdatei SYSTEM.JOBPOOL) hinterlegt hat, ENTER-JOB-Kommandos. Die Kommandos werden von JMP in eine Datei geschrieben. Von dort können sie entnommen, nach Bedarf modifiziert und dem System wieder übergeben werden.

| Anweisung | Bedeutung |
|---|---|
| CREATE-PROCEDURE-FILE | Prozedurdatei mit den ENTER-JOB-Kommandos erzeugen |
| OPEN-JOBPOOL-FILE | Job-Pool-Datei öffnen |
| CONVERT-JOBPOOL | Job-Pool-Datei in eine neueres oder älteres Format überführen |
| SHOW-JOBPOOL-STATUS | Informationen über den Inhalt einer Job-Pool-Datei ausgeben |
| END | JMP beenden |

Tabelle 31: Übersicht der JMP-Anweisungen

Eine ausführliche Beschreibung von JMP und Beispiele für rekonstruierte ENTER-JOB-Kommandos enthält das Handbuch „Dienstprogramme“ [15].

## 11.2 Task-Steuerung

Während die Job-Steuerung alle Jobs verwaltet, werden alle Tasks von der Task-Steuerung gesteuert und verwaltet.

Die zentralen Aufgaben der Task-Steuerung lassen sich wie folgt zusammenfassen:

– Tasks erzeugen
– Unterbrechungsanalyse und -bearbeitung
– Verwaltung der Registersicherstellungsbereiche
– Versorgung zentraler Tabellen des Ablaufteils
– Tasks aktivieren
– Tasks initiieren
– Tasks innerhalb und zwischen den Warteschlangen bewegen
– Tasks deaktivieren
– Tasks beenden

Unter dem Namen PRIOR sind nun jene Routinen der Task-Steuerung zu verstehen, die folgende Steuerungen übernehmen:

– die Steuerung von Tasks mittels Kategorien und Task-Scheduling-Prioritäten

– die Steuerung der System-Auslastung durch interne Regelfunktionen

Die Tasks sind allgemein Träger für unterbrechbare Abläufe in BS2000/OSD, wobei zwischen Benutzer- und Systemtasks unterschieden wird.

#### Benutzertask

Als Benutzertask werden alle – auch privilegierte – Benutzer-Jobs geführt, die von der Job-Steuerung der Task-Steuerung übergeben worden sind als:

– Dialogtask (Teilnehmerbetrieb; Task-Typ X'40')

– Batchtask (Batch-Betrieb; Task-Typ X'20')

#### Systemtask

Systemtasks nehmen die unterschiedlichsten Systemaufgaben wahr, wobei einige permanent sind (TCB im Klasse-1-Speicher vorgeneriert), die meisten jedoch dynamisch bei Bedarf erzeugt und wieder beendet werden.
Diesen Systemtasks sind vorgegebene Task-Identifier (TID) und TSNs zugeordnet, nach denen sie eindeutig identifiziert werden können.
Die wichtigsten Systemtasks, permanente und dynamisch (während Startup oder bei Bedarf) erzeugte, sind in den folgenden Übersichten zusammengefasst.

| TID | TSN | Aufgaben |
|---|---|---|
| 0001 0001 | TSC | Emergency Job Scheduler $SYSJS für die Job-Klasse $SYSJC und die Stream-Jobs |
| 0001 0003 | RMM | Prüfung des Hauptspeichers bei Startup |
| 0001 0005 | PT5 | Periodic Task 5: Prüft alle 10 Sekunden u.a. die LIFETIME (Wartezeitüberschreitung) beim Börsenmechanismus |
| 0001 0006 | PT6 | Periodic Task 6 des Job-Managements: Für Operationen, die zu einem bestimmten Zeitpunkt („AT“) durchgeführt werden sollen; Überprüfen der Repeat-Jobs (Intervall abgelaufen ?) |
| 0001 0007 | PGE | Paging Error Recovery Task |
| 0001 0008 | UCO | Universelle Konsole UCON: Verteilt die Konsolkommandos und Meldungen an die zuständige Instanz (Operator-Task für Operator-Normalkommandos; $CONSOLE-Anwendung für Operator-Spezial-kommandos); realisiert die Anwendung $CONSOLE |
| 0001 0009 | REK | Zuständig für Rekonfiguration |
| 0001 000A | VMM | Memory Management Task: Verwaltet die Slotpools; z.B für Stacks, Börsenkontrollblöcke |
| 0001 000B | MSG | Zuständig für Meldungswesen |
| 0001 000C | KTT | Konsol Treiber Task: Ein-/Ausgabe-Treiber für Konsolgeräte |

Tabelle 32: Permanente (preallocated) Systemtasks

| **TSN** | gehört zu den ... | **Aufgaben** |
|---|---|---|
| RUNT | | Operatortask: Zum Bearbeiten von aus einer RUN-Datei stammenden Operator-Normalkommandos |
| Xxxx | | Allgemeine Arbeitstasks des Systems, z.B. MSCF-Servertask, Import-/Exporttask, Operatortask (Bearbeitung und Ausführung der Operator-Normalkommandos) |
| CLOG | | Konsol-Logging (SYS.CONSLOG) |
| RP01 | | SRPM-Task; verwaltet den Benutzerkatalog des Home-Pubsets |
| RP02 | | SRPM-Task; verwaltet den Benutzerkatalog aller anderen Pubsets |
| HEL | | Verwaltet die Hardware-Fehlerdatei SYS.HEL.<datum>.<uhrzeit> |
| HERS | | HERS-Task; veranlasst HEL-Einträge bei Maschinenfehlern |
| SERS | | Verwaltet die Fehlerdatei SYS.SERSLOG.... |
| REPL | | Führt die Loggingdatei REPLOG und protokolliert Rep before image |
| DSSM<br>HTnn | Tasks der dyn. Subsystemverwaltung | Dynamic Subsystem Monitor<br>Hilfs-Tasks für DSSM |
| NKA<br>DM<br>TM<br>NKS | Tasks des Nucleus Device Management (NDM) | Betriebsmittelverwaltung<br>Disk-Monitor<br>Tape-Monitor<br>Betriebsmittelbelegung |
| ACCT<br>PACT | Tasks des Abrechnungssystems | Bearbeitet die Abrechnungsdatei<br>Task für die Funktion Periodisches Accounting |
| BCAM<br>BCAH<br>BCAF<br>BCAC<br><br>BCA0<br>BCAT | Tasks der Systemkomponente openNet Server | Nachrichtenpoolverwaltung; Verarbeitung der /BC-Kommandos<br>Holder-Task für Daten, die einen /BCEND überdauern<br>Dateizugriffstask (z.B. Lesen der Prozessor-Datei)<br>Kryptografie-Task; behandelt ver- und entschlüsselte Jobs bei Einsatz von IPSec<br>Transfer-Task für Datenverkehr über TCP/IP<br>Transfer-Task für Datenverkehr über andere Protokolle |
| DIAA | | TIAM-Task: erzeugt Dialog-Tasks<br>(für virtuelle Hosts auch: DIAB, DIAC, ...) |
| DBxx | | Tasks des Disk Access Buffers DAB |
| DRV | | Task des Software-Produkts DRV |
| SNAP | | SNAP-Task für die Realisierung der Funktion SNAP-Dump |

Tabelle 33: Wichtige, dynamisch erzeugte Systemtasks (Teil 1 von 2)

| **TSN** | gehört zu den ... | **Aufgaben** |
|---|---|---|
| S<br><br>SPAx<br>SPMG<br>xxxx<br><br>RSO<br>RSAx | Systemtasks von SPOOL | Spool-Scheduler; erzeugt Spool-Tasks und versorgt sie mit Jobs, zuständig für die Spool-Kommandos<br>Spool-Output-Writer; zuständig für SPOOLOUT<br>Spool-Message-Task; zuständig für Meldungsausgabe<br>Spool-Task (Device Controller Task); einem Spoolgerät zugeordnet (TSN wird vom System vergeben)<br>Remote SPOOL (RSO-Spool)<br>RSO-Output-Writer |
| M | | New Message Handler |
| WOPT | | Zuständig für die Hardware-Prüfprogramme unter der Benutzerkennung SERVICE |
| ISDS | | Verwaltung der ISAM-Pools |
| IOTD<br>IOTT | | Tasks zur Ein-/Ausgabeverwaltung bei Rekonfiguration von Platten (IOTD) und Bändern (IOTT) |
| JOBP | | Job-Pool-Controller-Task: Verwaltung der Job-Pool-Datei für Batch-Jobs |
| TSNA | | TSN-File-Manager: Verwaltung der Datei TSOJOIN.TSNN |
| TIME<br>TIMC | | Timer-Management: Uhrensynchronisation mit externem Zeitgeber zur Sommer-/Winterzeitumstellung |
| DRTx | | Domänen-Rekonfigurationstasks auf SQ-Servern |
| GSMT<br>GSRT | | Verwaltung des Globalspeichers<br>Rekonfiguration des Globalspeicher |
| MSCF<br>MSCM<br>WDGS | Systemtasks des Software-Produkts MSCF | Tasks für die Realisierung der MRS-Kopplung |
| FTCP | Task des Software-Produkts openFT | File Transfer Control Process (zentrale FT-Steuertask) |
| SM2G<br>SM2W<br>SM2U | Systemtasks des Software-Produkts openSM2 | Software-Monitor-Erfassungs-Task<br>openSM2-Schreib-Task<br>openSM2-Task für User-Messungen |
| SATP<br>PRxx<br><br>SATT | Systemtasks des Software-Produkts SECOS | SAT-Parameterdatei-Verwaltung<br>GUARDS-Server-Tasks; verwalten den Guards-Katalog eines Pubsets<br>SECURITY-AUDIT-TRAIL-Task für die SAT-Dateiverwaltung |

Tabelle 33: Wichtige, dynamisch erzeugte Systemtasks (Teil 2 von 2)

### Task-Kategorien

Jede Task ist einer Kategorie zugeordnet. Neben den vier Standardkategorien SYS, TP, DIALOG und BATCH unterstützt die Task-Steuerung 12 weitere Kategorien, deren Namen bei der Definition der Job-Klassen vereinbart werden.

Für die vier Standardkategorien gilt:

| | |
|---|---|
| SYS | nur für Systemtasks |
| TP | für Transaktions-Tasks (Teilhaberbetrieb) |
| DIALOG | für Dialogtasks (Teilnehmerbetrieb) |
| BATCH | für Batch-Tasks (Batch-Betrieb) |

Zusätzlich gibt es vier Task-Attribute, deren Namen identisch sind mit den Standardkategorienamen SYS, TP, DIALOG und BATCH. Den Task-Attributen sind spezielle, für das Task-Scheduling wichtige Ablaufparameter zugeordnet.

Das Task-Attribut TP zeichnet sich gegenüber den anderen Task-Attributen durch eine speziell auf die Bedürfnisse des Teilhaberbetriebs optimierte Hauptspeicher-Verwaltung aus. Dieses Attribut kann entweder durch Definition in der den Jobs zugeordneten Job-Klassen oder durch Aufruf des TINF-Makros (siehe Handbuch „Makroaufrufe an den Ablaufteil“ [30]) erlangt werden. Die dazu notwendige Berechtigung muss im Benutzerkatalog eingetragen sein.

Das Konzept der Task-Kategorien ist nur der Systemverwaltung (im Ausnahmefall auch dem Operator) über die Kommandoschnittstelle zugänglich.
Mit dem MODIFY-TASK-CATEGORIES-Kommando spezifiziert die Systembetreuung die Wichtigkeit der Kategorien untereinander für die Aktivierung (= Zuteilung des Betriebsmittels Hauptspeicher).

Zu diesem Zweck werden folgende Kategorie-Attribute vereinbart:

| | |
|---|---|
| WEIGHT | Wichtigkeit (Gewicht) |
| MINMPL | Minimum an aktiv zu haltenden Tasks |
| MAXMPL | Maximum an aktiv zu haltenden Tasks |
| IO-PRIO | I/O-Priorität aller Tasks dieser Kategorie |

Ob eine Task-Aktivierung, -Deaktivierung oder -Verdrängung durchgeführt wird, ist neben den Kategorie-Attributen noch abhängig von der System-Auslastung und der Priorität der betroffenen Task.

Da einer Kategorie in der Regel mehrere Tasks zugeordnet sind, ist die Kategoriensteuerung immer **gruppenspezifisch** zu betrachten. Will man eine Beurteilung der Auswirkungen der Kategoriensteuerung versuchen, so ist zwischen den folgenden Situationen zu unterscheiden:

- Im Unterlast- bzw. Normallast-Bereich hat die Kategoriensteuerung keinen nennenswerten Einfluss.
- Im Vollast- bzw. Überlast-Bereich (also bei Betriebsmittelengpässen) hat sie eine starke Wirkung und dient zur Lastbegrenzung, d.h. weniger wichtige Kategorien werden in den Hintergrund gedrängt.

Mit dem Kommando MODIFY-TASK-CATEGORIES, Operand IO-PRIORITY, kann die I/O-Priorität aller Tasks der angegebenen Katagorie bestimmt werden. Dieser Wert wird nur ausgewertet, wenn das Subsystem IORM gestartet ist. Die Voreinstellung *NONE bedeutet, dass alle Tasks dieser Kategorie eine aus der Task-Priorität abgeleitete I/O-Priorität erhalten (implizite I/O-Priorität). Die I/O-Prioritäten werden ausschließlich vom Subsystem IORM zur task-spezifischen Steuerung von Ein-/Ausgaben ausgewertet und zwar mit Ablaufsteuerung sowohl durch PRIOR als auch durch das Subsystem PCS.

Mit dem Kommando MOVE-TASK-TO-CATEGORY kann die Systembetreuung die Zuordnung einer Task zu einer Kategorie ändern, wenn etwa eine andere (bessere) Bedienung dieser Task oder auch eine Entlastung einer Kategorie erreicht werden soll (mit oder ohne Einsatz von PCS).
Zielkategorien müssen JMS-Kategorien und dürfen keine Folgekategorien sein. Die Kategorie SYS wird weder als Ausgangs- noch als Zielkategorie unterstützt.

Die Wahl geeigneter Kategorienwerte ist häufig ein Problem. Deshalb empfiehlt es sich (vor allem für große Server) über längere Zeiträume hinweg Performance-Untersuchungen durchzuführen. Die dazu notwendigen Daten liefert der Software Monitor openSM2 (siehe Handbücher „openSM2“ [51] und „SM2-PA“ [52]). Weitere Informationen enthält das „Performance-Handbuch“ [39].

| **Kommando** | **Bedeutung** |
|---|---|
| ADD-USER | Eintrag im Benutzerkatalog erstellen mit der Vereinbarung, ob die Tasks des Benutzers deaktiviert werden dürfen oder nicht |
| CANCEL-JOB | Benutzerauftrag abbrechen |
| CHANGE-TASK-CPU-LIMIT | Maximale CPU-Zeit eines Batch-Jobs erhöhen |
| CHANGE-TASK-PRIORITY | Job- oder Task-Priorität ändern |
| ENTER-JOB | Kommandofolge, die in einer ENTER-Datei als Batch-Job gespeichert ist, unter Angabe der Task-Priorität starten |
| FORCE-JOB-CANCEL | Benutzerauftrag abbrechen |
| HOLD-TASK | Benutzerauftrag anhalten |
| MODIFY-TASK-CATEGORIES | Verteilung der Betriebsmittel CPU, Hauptspeicher und I/O-Prozessor auf die einzelnen Kategorien regeln, Dringlichkeiten der Kategorien festlegen und I/O-Prioritäten für alle Tasks einer Kategorie vergeben |
| MOVE-TASK-TO-CATEGORY | Kategorie-Zuordnung einer Task ändern |
| RESUME-TASK | Angehaltenen Benutzerauftrag freigeben |
| SET-LOGON-PARAMETERS | Dialog-/Batch-Job unter Angabe der Task-Priorität einleiten |
| SHOW-JOB-STATUS | Informationen über einen Benutzer-Job anfordern |
| SHOW-SYSTEM-STATUS | Informationen über Job-Klassen und Job-Streams anfordern |
| SHOW-USER-ATTRIBUTES | Maximale Task-Priorität abfragen |
| SHOW-USER-STATUS | Informationen über Gruppen von Benutzer-Jobs anfordern |
| **Makro** | **Bedeutung** |
| PASS | eine Sekunde warten |
| TINF | Task-Attribute lesen und ändern |
| TSPRIO | Runprioritäten ausgeben |
| VPASS | warten |

Tabelle 34: Schnittstellenübersicht zur Task-Steuerung

### 11.2.1 Prioritätenkonzept und Warteschlangen

Die Ausführungsprioritäten lassen sich wie folgt einteilen:

| | |
|---|---|
| Prioritäten 0 - 29 | Prioritäten für Systemtasks |
| Prioritäten 30 - 127 | Feste Task-Scheduling-Prioritäten |
| Prioritäten 128 - 255 | Variable Task-Scheduling-Prioritäten |

Mit Ausnahme der Systemprioritäten werden die Prioritäten bei der Job-Klassen-Definition und benutzerspezifisch im Benutzerkatalog festgelegt.

Gibt der Benutzer im SET-LOGON-PARAMETERS- bzw. ENTER-JOB-Kommando eine Priorität an, dann wird diese sowohl in der dem Benutzer zugewiesenen Job-Klasse als auch im Benutzerkatalog geprüft.

Die Priorität einer Task wird sowohl bei der Aktivierung als auch bei der Initiierung (Zuteilung des Betriebsmittels CPU) berücksichtigt.

Über den Systemparameter ETMFXLOW gibt es die Möglichkeit, auch einen unteren Bereich von festen Prioritäten einzurichten (siehe Seite 742).

#### Variable Prioritäten

Charakteristisch für die variablen Prioritäten ist die dynamische Anpassung der Priorität mit dem HRN-Algorithmus (= Highest-Response ratio Next).

HRN basiert auf dem Verhältnis „Verweilzeit / verbrauchte CPU-Zeit“ unter Berücksichtigung der Start-Priorität beim SET-LOGON-PARAMETERS bzw. ENTER-JOB-Kommando oder einer extern zugewiesenen Priorität (Kommando CHANGE-TASK-PRIORITY).

Der HRN-Algorithmus bewirkt, dass Tasks, die wenig CPU-Leistung aufnehmen, und Ein-/Ausgabe-intensive Tasks bevorzugt behandelt werden, ohne dabei rechenintensive Tasks extrem stark zu benachteiligen.
Darüber hinaus wird eine angemessene Versorgung auch für Tasks mit niedriger Start-Priorität gewährleistet.

Die variable Priorität wird zu folgenden Zeitpunkten neu berechnet:

- bei jeder Aktivierung
- bei jedem PEND/UNPEND nach Q5
- bei Ablauf der Mikrozeitscheibe (einem von der CPU-Geschwindigkeit und vom unmittelbar zurückliegenden E/A-Verhalten der Task abhängigen Zeitquantum)
- periodisch (1/Sekunde)
- bei Absetzen des Kommandos MODIFY-TASK-CATEGORIES
- bei Absetzen des Kommandos CHANGE-TASK-PRIORITY

### Feste Prioritäten

Die festen Prioritäten ändern sich nicht.

Für die betroffene Task stellt die feste Priorität eine starke Bevorzugung dar.
Feste Prioritäten sind für Anwendungen mit extremen Realzeitanforderungen entwickelt worden. Durch die Vergabe einer festen Priorität kann eine Performance-Verbesserung erreicht werden, jedoch unter Berücksichtigung folgender Punkte:
Feste Prioritäten engen den Entscheidungsspielraum des Systems stark ein. Zur Erzielung eines positiven Effektes müssen daher die Betriebsmittelanforderungen sämtlicher Tasks bekannt sein. Zur Aktivierung anstehende Tasks mit fester Priorität führen bei hoch ausgelasteten Betriebsmitteln zu einer sofortigen Verdrängung anderer Tasks, wodurch sich eine Überlastsituation ergeben kann. Das wiederum hat letztendlich negative Auswirkungen auf die Performance.

Feste Prioritäten sollten nur nach genauer Analyse der Last, der Betriebsmittel-Auslastung und zusammen mit lastbegrenzenden Maßnahmen (wie z.B. Beschränkung des Multiprogrammierungs-Grades der Kategorien durch den Operanden MAXIMUM-ACTIVE-TASKS im Kommando MODIFY-TASK-CATEGORIES) vergeben werden.

Zusammenfassend lässt sich Folgendes feststellen:

- Prioritäten, egal welche, beeinflussen die Task-Reihenfolge in den Warteschlangen.
- Jede Priorität, die besser ist als die „normale Priorität“ 255, beeinflusst das System unabhängig von der Lastsituation.
- Auf Grund der gesteigerten Bedeutung der Priorität kann die Systembetreuung durch Prioritätsänderung die Performance einzelner Tasks sehr leicht beeinflussen. Vor allem für leistungsschwächere Systeme mit nur wenigen Benutzertasks bietet sich an, die Tasks mit variablen Prioritäten unterschiedlicher Größenordnung zu belegen. Damit kann erreicht werden, dass jede Task die Leistung erhält, die ihrer Wichtigkeit entspricht.
- Arbeiten fast alle Tasks mit Priorität 255, genügt ein kleiner Prioritätsunterschied (z.B. 5), um eine Task deutlich zu bevorzugen. Bei einem größeren Prioritätsspektrum muss der Prioritätenunterschied entsprechend größer gewählt werden.
  Die Variation der Priorität sollte nach Last z.B. in 5er- oder 10er-Schritten vorgenommen werden, bis das gewünschte Performanceziel erreicht ist. In der Regel ist es unnötig, feste Prioritäten zu vergeben, da mit mittleren variablen Prioritäten (etwa 200) bereits gute Performancewerte erreicht werden können.
  Größere Prioritätsschritte sind bei einem permanent hohen Task-Aufkommen bzw. bei überwiegend Ein-/Ausgabe-intensiven Tasks notwendig.
- Die Zuteilung der Betriebsmittel Hauptspeicher und CPU an Benutzer- und Systemtasks wird entsprechend dem vorgegebenen Kategorien- und Prioritätenkonzept gesteuert.

### Warteschlangen

Im System vorhandene Tasks können sich in fünf Zuständen befinden:

- die Task belegt eine CPU
- die Task ist aktiv und ablaufbereit
- die Task ist aktiv, aber nicht ablaufbereit
- die Task ist inaktiv und ablaufbereit
- die Task ist inaktiv und nicht ablaufbereit

Die notwendige Basis bietet das Warteschlangensystem der Task-Steuerung, das die Tasks in Abhängigkeit des jeweiligen Zustandes in eine der folgenden Warteschlangen einreiht.

| Q0 | Laufende Task | Task in Control |
|---|---|---|
| Q1 | Warten auf Zuteilung einer CPU | aktiv<br>und ablaufbereit |
| Q2 | Schreibtasks von openSM2 | aktiv<br>und<br>nicht ablaufbereit |
| Q3 | Warten auf Seitenwechsel | |
| Q4 | Warten bei schneller Ein-/Ausgabe,<br>Explizite Synchronisationsfunktionen (VPASS, SOLSIG, REVNT, Locks und Auftragsbeziehungen im System),<br>Warten auf neue Eingabe für TP-Tasks | |
| Q5 | pro Kategorie: Warten auf Aktivierung | inaktiv<br>und<br>ablaufbereit |
| Q6 | pro Kategorie: Warten auf Zulassung durch PCS<br>(nur falls PCS eingeschaltet ist) | |
| Q10 | Warten im HOLD-Zustand | inaktiv<br>und<br>nicht ablaufbereit |
| Q11 | Wartende Systemtasks | |
| Q12 | Warten bei langdauernder Ein- und Ausgabe und langdauernden Synchronisationsfunktionen insbesondere Terminal-Ein-/Ausgabe im Dialog | |
| Q13 | Warten n Sekunden (PASS/VPASS) | |

Tabelle 35: Warteschlangen der Task-Steuerung

Wartet eine Task nicht auf eine Eingabe, so hat sie das primäre Ziel, die CPU zu belegen. Das setzt in der Regel mehrere Zustandswechsel voraus.

Eine Benutzertask (z.B. der Kategorie Batch) befindet sich z.B. nach einem HOLD-TASK-Kommando in der Q10.
Sobald die Systembetreuung die Task mit dem Kommando RESUME-TASK wieder freigibt, verändert sie ihren Zustand.

Die Q10 wird verlassen und die Task wird in die Q5 (Subqueue für die jeweilige Kategorie, z.B. Batch) eingereiht. Dort wartet sie darauf, dass PRIOR über die Aktivierungsentscheidung auf diese Task zugreift und sie in die Q1 bringt und danach in die Q0.

Diese Zustandswechsel werden durch die PEND-/UNPEND-Routinen realisiert:
Die UNPEND-Routinen bringen die Task in Richtung CPU, die PEND-Routinen bringen die Task von einer hochwertigen Warteschlange in eine niederwertige Warteschlange.

## 11.2.2 Zuteilung von Ressourcen

Die Task-Steuerung muss eine Reihe von Entscheidungen treffen, wenn eine Task zum Ablauf kommen soll. Der Verwaltung der Betriebsmittel kommt dabei eine zentrale Bedeutung zu.

Erhält eine Task das Recht, Hauptspeicher zu belegen, dann heißt dies, dass die Task aktiviert wird.
Im Anschluss daran muss der Task eine CPU zugewiesen werden, d.h. die Task wird initiiert.

Kriterien für die Aktivierungsentscheidung sind:

- der Multiprogrammierungs-Grad pro Kategorie
- die Priorität
- Betriebsmittelauslastung (Hauptspeicher, CPU, Seitenwechsel)
- geleistete Systemdienste (CPU-Zeit, Anzahl der Ein- und Ausgaben)

### Zeitscheiben

Die Zeitscheiben dienen der Task-Steuerung, um die Betriebsmittel CPU und Hauptspeicher allen Tasks möglichst gleichmäßig und gleichberechtigt zur Verfügung zu stellen:

- Mikrozeitscheibe (MTS) zur optimalen Nutzung der CPU
- Systemdienstzeitscheibe (SSTS) für sinnvolle Zuteilung des Hauptspeichers

*Mikrozeitscheibe*

Die Mikrozeitscheibe MTS ist die maximale Zeit, die eine Task das Betriebsmittel CPU ohne Unterbrechung beanspruchen kann, wenn keine Task mit höherer Priorität auf Zuteilung der CPU wartet.
Spätestens nach Ablauf der Mikrozeitscheibe erfolgt eine Unterbrechung, um die Blockierung der CPU durch eine Task zu verhindern. Besaß die Task eine variable Priorität (128-255), wird diese entsprechend des E/A-Verhaltens und der CPU-Nutzung der Task verändert.

*Systemdienstzeitscheibe*

Die Systemdienstzeitscheibe SSTS dokumentiert keine absolute Zeit, sondern entspricht einem Quantum an Systemdiensten (CPU-Nutzung, E/A-Rate), die eine aktive Task maximal beanspruchen darf. Nach Ablauf der Systemdienstzeitscheibe wird die Task deaktiviert, wenn Tasks der gleichen Kategorie auf Aktivierung warten.

### Zuteilung des Betriebsmittels Hauptspeicher

- Aktivieren einer Task

  Die Aktivierung einer Task ist nur möglich, wenn die interne Kontrollinstanz, die ACF (Activate-Control-Function), dies erlaubt. D.h. ACF misst in periodischen Abständen die Auslastung der Betriebsmittel Hauptspeicher und CPU sowie die Seitenwechsel-Aktivitäten, bzw. wenn eine Folge von Aktivierungen durchgeführt werden soll, nur die Auslastung des Hauptspeichers.
  Je nach Betriebsmittelauslastung und definiertem Kategorien- und Prioritätenkonzept erfolgt dann anhand einer Matrix die Entscheidung, ob eine weitere Aktivierung zulässig ist oder ob eine Zwangsdeaktivierung, eine Verdrängung oder gar nichts erfolgen soll.

  Die Vorgehensweise des PRIOR bei der Aktivierung einer Task soll im Folgenden an einem Beispiel gezeigt werden:

  Insgesamt sind aktiv: 19 TP-, 9 DIALOG- und 5 BATCH-Tasks

  Auf Aktivierung warten: 2 TP-, 2 DIALOG- und 2 BATCH-Tasks

  Die Systembetreuung hat die Kategorien wie folgt definiert, der Operand IO-PRIORITY ist auf *NONE gesetzt (Voreinstellung):

```
/MODIFY-TASK-CATEGORIES CATEGORY-NAME=DIALOG,-
        WEIGHT-CODE=100,MINIMUM-ACTIVE-TASKS=25,MAXIMUM-ACTIVE-TASKS=35

/MODIFY-TASK-CATEGORIES CATEGORY-NAME=TP,-
        WEIGHT-CODE=500,MINIMUM-ACTIVE-TASKS=30,MAXIMUM-ACTIVE-TASKS=40

/MODIFY-TASK-CATEGORIES CATEGORY-NAME=BATCH,-
        WEIGHT-CODE=3,MINIMUM-ACTIVE-TASKS=5,MAXIMUM-ACTIVE-TASKS=15
```

  Die Auswahl-/Aktivierungsentscheidung trifft die Task-Steuerung in zwei Stufen.

  1. Auswahl der Kategorie, aus der eine Task aktiviert werden soll.
  2. Innerhalb dieser Kategorie wird die Task ausgewählt, die die höchste Priorität hat bzw. bei gleicher Priorität wird die Task aktiviert, die in der Warteschlange an erster Stelle steht.

| **Für die Auswahl der Kategorie sind folgende Entscheidungen notwendig** | **Bezogen auf das Beispiel bedeutet dies** |
|---|---|
| – Es werden nur die Kategorien berücksichtigt, für die **mindestens eine Task auf Aktivierung wartet** | – die Bedingung trifft für alle Kategorien zu |
| – Bevorzugt ausgewählt werden unter diesen Kategorien<br>1. alle Kategorien (sofern vorhanden), die MINMPL noch nicht erreicht haben. MAXMPL ist hier ohne Bedeutung.<br>2. wenn alle Kategorien MINMPL erreicht haben: die Kategorien, die MAXMPL noch nicht erreicht haben.<br>Sind solche bevorzugten Kategorien vorhanden, werden alle anderen Kategorien von der weiteren Auswahl ausgeschlossen.<br>z=0<br>Anzahl aktiver Tasks < MIN<br>z=MIN<br>MIN ≤ Anzahl aktiver Tasks < MAX<br>z=MAX<br>Anzahl aktiver Tasks > MAX | – die Bedingung trifft für die Kategorien TP und DIALOG zu<br>Begründung:<br>z=0 für TP (19 < 30)<br>z=0 für DIALOG ( 9 < 25)<br>z=5 für BATCH (MIN=5) |
| – Von diesen Kategorien wird nun die Kategorie für die Aktivierung ausgewählt, für die der niedrigste Index ermittelt wird. Die Indexberechnung erfolgt nach dem Algorithmus:<br>$Index = \frac{NAT + 1 - z}{WT}$<br>Dabei bedeutet:<br>NAT Anzahl der aktiven Task der Kategorie, für die der Indexberechnet wird<br>z Zustandsanzeige<br>WT Wichtigkeit (Gewicht) | – Für die Kategorie TP und DIALOG muss die Indexberechnung erfolgen.<br>$Index_{TP} = \frac{19 + 1 - 0}{500} = \frac{20}{500} = 0,04$<br>$Index_{DIA} = \frac{9 + 1 - 0}{100} = \frac{10}{100} = 0,10$ |
| – Bei Index-Gleichheit wird in der Reihenfolge TP - DIALOG - BATCH aktiviert. | |

Ergebnis: Es wird die höchstpriore Task der Kategorie TP aktiviert.

Ausnahme: Tasks mit fester Priorität werden bevorzugt aktiviert.

- Deaktivieren einer Task

  Das Deaktivieren von aktiven Tasks verfolgt den Zweck, ablaufbereiten inaktiven Tasks das Betriebsmittel Hauptspeicher zur Verfügung zu stellen.
  Es geschieht, wenn für die aktiven Tasks

  – eine weitere Verarbeitung nicht möglich ist
    (z.B. durch programmgesteuerte Warteaufrufe, Warten auf Eingabe von Datensichtstation im Teilnehmerbetrieb)

  – die Zeit beim Warten auf Ereignisse überschritten wird
    (z.B. Warten auf Eingabe von Datensichtstation im Teilhaberbetrieb)

  – ein bestimmter Umfang von System-Diensten
    (z.B. CPU-Zeit und Anzahl der Ein-/ Ausgaben) in Anspruch genommen wurden.

  *Ausnahme*

  Die Deaktivierung unterbleibt, wenn die Systembetreuung im Benutzerkatalog für den Benutzer vereinbart hat, dass seine Tasks nicht deaktiviert werden dürfen (siehe Kommando ADD-USER, Operand PRIVILEGE=INHIBIT-DEACTIVATION).

- Zwangsdeaktivieren einer Task

  Wenn die ACF-Funktion im Rahmen ihrer Messungen zur Auslastung der Betriebsmittel Engpässe feststellt, kommt es zur Zwangsdeaktivierung einer Task.
  Da die Kategorie-Merkmale MIN, MAX und WT für die Zwangsdeaktivierung herangezogen werden, kann die Systembetreuung mit einer entsprechenden Voreinstellung dieser Werte dafür sorgen, dass nur Tasks weniger wichtiger Kategorien zwangsdeaktiviert werden.

  *Ausnahmen*

  – Tasks mit fester Priorität
  – Tasks im Funktionszustand TPR
  – Tasks, die einen Lock halten
  – Tasks, für deren Benutzerkennung PRIVILEGE=INHIBIT-DEACTIVATION vereinbart wurde (Kommando ADD-USER), können nicht zwangsdeaktiviert werden.

- Verdrängen einer Task

  Zu einer Verdrängung kann es kommen, wenn eine Aktivierungsanforderung vorliegt und die ACF-Funktion auf Grund festgestellter Betriebsmittelengpässe keine Aktivierung zulässt. Es gibt zwei Möglichkeiten der Verdrängung:

  1. Eine aktive Task wird von einer inaktiven Task einer anderen Kategorie verdrängt.
  2. Die aktive und inaktive Task gehören der gleichen Kategorie an.

  Die Verdrängungsrate ist umso höher, je langfristiger der Überlastzustand (= Engpass an Betriebsmitteln) im System ist.

- Preemption-Control-Function (PCF)

  PCF überwacht periodisch die Verdrängungsrate.
  Meldung: `EXC0455 TASK PREEMPTION LEVEL=i`

  `i` ist eine Ziffer von 0 bis 3 und bezeichnet den Grad der Verdrängungsrate.

| Grad | Auswirkungen im System | Maßnahme des Operators |
|---|---|---|
| 0 | Normaler Systemablauf | Keine |
| 1, 2 | Kurzzeitige Überlastzustände | Keine |
| 3 | Langfristige Überlastzustände<br>Der MAX-MPL-Wert einer Katagorie wird nicht überschritten. Das System hält Hauptspeicher für Privilegierungen frei. | Reduzierung der MIN-MPL-Werte von Katagorien |

*Maßnahmen zur Verhinderung zu hoher Verdrängungsraten*

- Ändern Sie die MIN-MPL- und MAX-MPL-Werte für einzelne Katagorien so ab, dass weniger Aktivierungsanforderungen entstehen.
- Vergrößern Sie den Hauptspeicher oder reduzieren Sie die Last.

### Zuteilung des Betriebsmittels CPU

Jede Task wartet nach erfolgter Aktivierung auf Zuteilung des Betriebsmittels CPU (Zentral-prozessor), d.h. auf **Initiierung**.

Hat die Task das Betriebsmittel CPU erhalten, kann es auf Grund folgender Unterbrechungsursachen zu einer **Deinitiierung** kommen:

- Task will auf Synchronisationsereignis warten
- Warten auf Beendigung einer Ein-/Ausgabe-Operation
- Seitenwechsel vornehmen
- Ablauf der Mikrozeitscheibe

Auch auf Initiierungsebene kann es zu einer **Verdrängung** (Preemption) kommen.
Dies erfolgt dann, wenn eine andere aktive und ablaufbereite Task mit höherer Priorität auf Initiierung wartet. Das ist z.B. der Fall, wenn für eine Task mit höherer Priorität eine Ein-/ Ausgabe-Operation abgeschlossen ist.

### Zuteilung von I/O-Ressourcen

Die Zuteilung von I/O-Ressourcen ist abhängig von der I/O-Priorität des Tasks. Die Steuerung übernimmt IORM, siehe Abschnitt „IORM: Steuerung von I/O-Ressourcen“ auf Seite 254 und Handbuch „Dienstprogramme“ [15].

### 11.2.3 TANGRAM: Verwaltung von affinen Task-Gruppen

Das Subsystem TANGRAM (Task and Group Affinity Management) plant für affine Task-Gruppen unter Berücksichtigung ihres Leistungsbedarfs die Zuordnung auf Prozessoren. Bei bestimmten Multitask-Anwendungen gibt es Gruppen von Tasks, die häufig auf größere Mengen gemeinsamer Daten schreibend zugreifen. Diese Task-Gruppen werden als „affine Task-Gruppen“ bezeichnet.
Um auf Multiprozessoren die Verluste durch Cohits in den CPU-Caches zu reduzieren, ist es günstig, solche Task-Gruppen auf eine Teilmenge der Prozessoren einzuschränken.

Das Einrichten der Task-Gruppen erfolgt mit dem Makro TINF (siehe Handbuch „Makroaufrufe an den Ablaufteil“ [30]). Die Verwaltung der Taskgruppen übernimmt das bei „System Ready“ automatisch gestartete Subsystem TANGBAS und zwar unabhängig davon, ob die Regelungsfunktion TANGRAM gestartet ist.

Das Subsystem TANGRAM misst die Auslastung der einzelnen Prozessoren, die Leistungsaufnahme der angemeldeten Task-Gruppen sowie der restlichen Tasks und legt jeweils für ein Regelintervall die für die Initiierung einer Task erlaubten Prozessoren fest.

Die An- und Abmeldung von Tasks zu Gruppen sowie die Prozessorzuordnung der einzelnen Gruppen kann mit COSMOS-Hooks protokolliert werden. COSMOS ist ein optionaler Bestandteil von openSM2 (siehe Messprogramme im Handbuch „openSM2“ [51]).

Der durch den Einsatz von TANGRAM erzielbare Performance-Gewinn ist abhängig vom Server (Multiprozessorgrad, Cache-Architektur), vom Lastprofil (Anteil und Strukturierung der TP-Anwendungen) und von der Server-Auslastung. Die BS2000-TP-Produkte openUTM, UDS/SQL und SESAM/SQL richten TANGRAM-Task-Gruppen ein. Multitask-Anwendungen sollten daraufhin überprüft werden, ob das Einrichten von TANGRAM-Task-Gruppen einen Performance-Gewinn bringt.

Die Arbeitsweise von TANGRAM kann durch die Angabe von Parametern in der Parameterdatei SYSSSI.TANGRAM.<version> gesteuert werden.
Die angegebenen Voreinstellungen treten in Kraft, wenn einer der Parameter nicht in der Parameterdatei gefunden wird oder keine Parameterdatei zur Verfügung steht. Im zweiten Fall wird außerdem folgende Konsolmeldung ausgegeben:

```
NTG0002 Warning: Opening of TANGRAM parameter file (&00) failed.
        DMS-Error-Code (&01). Processing continued with default parameters.
```

Um Parameter zu verändern, muss TANGRAM angehalten und mit der geänderten Parameterdatei neu gestartet werden. Bei diesem Vorgang bleiben alle eingerichteten Task-Gruppen erhalten.

Die folgenden Parameter steuern die Arbeit von TANGRAM. Die Reihenfolge der Parameter in der Parameterdatei ist beliebig, es muss jedoch jeder Parameter in einer eigenen Zeile stehen.

**PERIOD=<integer 1..100>**
Bestimmt den Abstand zwischen zwei aufeinander folgenden Läufen der periodischen TANGRAM-Routinen in Sekunden. Voreinstellung: 10 Sekunden

**CLEARENCE=<integer 0..100>**
Gibt einen Wert zur Berechnung der max. CPU-Auslastung einer Task-Gruppe in Prozent an. Die Auslastung wird folgendermaßen berechnet: Auslastung = (1-wert).
Voreinstellung: 20 %, d.h. jede CPU sollte maximal zu 80 % von einer einzelnen Task-Gruppe belegt sein.

**THRESHOLD=<integer 0..100>**
Bestimmt die Schwelle, ab welcher Gesamtauslastung auf allen Prozessoren eine Task-Gruppe aktiv am Verfahren der Prozessor-Zuordnung durch TANGRAM teilnimmt.
Voreinstellung: 10 %

## 11.3 Zeitlimitierungen in BS2000/OSD

In BS2000/OSD existieren drei verschiedene Zeitlimitierungen, die den Benutzer im Teilnehmerbetrieb betreffen: Eine benutzerkennungs-spezifische, eine task-spezifische und eine programmlauf-spezifische Zeitbegrenzung.

Eine benutzerkennungs-spezifische Zeitbegrenzung legt die Systembetreuung im Benutzereintrag fest. Von diesem festgelegten Zeitkontingent rechnet das Abrechnungssystem von BS2000/OSD bei Task-Beendigung die verbrauchte CPU-Zeit ab.

Für eine Task kann beim Starten die maximale CPU-Zeit (Task-Time-Limit: TTL) angegeben werden.

- Für Batch-Jobs gilt:
  Die maximal angebbare CPU-Zeit ist das Minimum aus dem Zeitkontingent des Benutzereintrags und dem Maximalwert aus der Job-Klassen-Definition.
- Für Dialog-Jobs gilt:
  Die maximale CPU-Zeit wird mit dem Maximalwert aus der Job-Klassen-Definition vorgegeben, sofern der CPU-Zeit-Wert im Benutzereintrag noch größer Null ist.

Ohne Angabe einer maximalen CPU-Zeit wird der Standardwert aus der Job-Klassen-Definition oder gegebenenfalls das kleinere Zeitkontingent des Benutzereintrags als maximale CPU-Zeit der Task übernommen. Benutzer, die im Benutzereintrag das Privileg No-Time-Limit besitzen, und die Benutzerkennung TSOS können Tasks ohne Zeitbegrenzung starten (NTL-Tasks).

Da das Zeitkontingent im Benutzereintrag erst bei Beendigung der Task aktualisiert wird, kann die maximale CPU-Zeit (und ggf. die tatsächlich verbrauchte) mehrerer Tasks einer Benutzerkennung zusammen das Zeitkontingent überschreiten.

Bei dem Laden oder Starten eines Programms kann für den jeweiligen Programmlauf eine maximale Programmlaufzeit angegeben werden (Program-Time-Limit: PTL).

Bezüglich der maximalen CPU-Zeit der Task (TTL) und der maximalen Programmlaufzeit (PTL) verhalten sich Dialog- und Batch-Jobs unterschiedlich:

### TTL bzw. PTL im Batch-Job erreicht

Hat der Batch-Job die maximale CPU-Zeit verbraucht und ist kein Programm geladen und war TTL < 300 Sekunden, so wird die Task mit folgender Meldung an der Bedienstation beendet:

```
EXC0736 ABNORMAL TASK TERMINATION.
        ERROR CODE '(CMD1011)': /HELP-MSG (CMD1011)
```

Hat der Batch-Job die maximale CPU-Zeit verbraucht und ist ein Programm geladen oder ist TTL ≥ 300 Sekunden, wird folgende Meldung an der Bedienstation ausgegeben:

```
EXC0070 Batch job with TSN (&00) has reached time runout.
```

Wenn der Systemparameter DIATTL=Q gesetzt ist (siehe Seite 493), dann wird der Batch-Job beendet.

Sonst wird der Job jetzt für 20 min. angehalten, falls ein Programm geladen ist. Falls kein Programm geladen ist, wird das TTL des Jobs um 5 Minuten erhöht. Während dieser Wartezeit (5 CPU- bzw. 20 Zeit-Minuten) kann die Systembetreuung mit dem Kommando CHANGE-TASK-CPU-LIMIT die maximale CPU-Zeit des Batch-Jobs erhöhen und verhindern, dass wichtige Batch-Jobs wegen Zeitüberschreitung vorzeitig beendet werden.

Wird die maximale CPU-Zeit des Batch-Jobs innerhalb der Wartezeit von 5 bzw. 20 Minuten nicht erhöht, wird der Batch-Job (endgültig) beendet. Dabei sind folgende Fälle zu unterscheiden:

1. Wurde das TTL bzw. PTL während eines Programmlaufes erreicht, wird die Meldung `EXC0072` ausgegeben oder ggf. eine STXIT-Routine abgearbeitet, die maximal weitere 30 CPU-Sekunden verbrauchen darf.
   Nach Abarbeitung der STXIT-Routine bzw. spätestens nach 30 CPU-Sekunden wird das Programm mit der Meldung `EXC0073` beendet.

   Anschließend wird die Task beendet (TTL) oder Spin-Off ausgelöst (PTL).
   Ist eine Fehlerbehandlung vorgesehen, beginnt die weitere Bearbeitung bei dem nächsten SET-JOB-STEP- bzw. IF-BLOCK-ERROR-Kommando. Anderenfalls endet die Bearbeitung beim nächsten EXIT-JOB- bzw. LOGOFF-Kommando (PTL).

   ```
   EXC0072 TIME LIMIT FOR TASK/PROGRAM EXCEEDED. PROGRAM TERMINATED
   EXC0073 MAXIMUM PROGRAM RUNTIME EXCEEDED. 'STXIT' ROUTINE OR
           PROGRAM RUN USED UP 30 SECONDS.
   ```

2. Wurde das TTL während der Abarbeitung von BS2000-Kommandos erreicht, wird die Task mit folgender Meldung beendet:

   ```
   EXC0736 ABNORMAL TASK TERMINATION.
           ERROR CODE '(CMD1011)': /HELP-MSG (CMD1011)
   ```

### TTL bzw. PTL im Dialog-Job erreicht

Im Normalfall ist die CPU-Zeit einer Dialogtask nicht begrenzt. Die Systembetreuung kann jedoch mit dem Systemparameter DIATTL (DIAlog Task Time Limit) die CPU-Zeit in Abhängigkeit vom TTL begrenzen. DIATTL kann die Werte Y, N und Q annehmen:

1. DIATTL=N (Standard)
   Die Dialogtask wird mit Task-Time-Runout nicht beendet.

   Bei Überschreiten des TTL wird, wenn kein Programm geladen ist, die Meldung `EXC0067` ausgegeben und der Benutzer kann 30 CPU-Sekunden weiterarbeiten, bis sich der Vorgang wiederholt.

   ```
   EXC0067 CPU TIME SPECIFIED IN /SET-LOGON-PARAMETERS EXCEEDED.
           TASK CONTINUED
   ```

   Wenn ein Programm geladen ist, gilt bei Überschreiten des TTL bzw. PTL:

   – Im Prozedur-Modus wird die Meldung `EXC0068` ausgegeben und die Ausführung fortgesetzt. Die Zeitbegrenzung wird um 30 CPU-Sekunden hochgesetzt, d.h. danach wiederholt sich der Vorgang.

   ```
   EXC0068 CPU TIME SPECIFIED AT PROGRAM START EXCEEDED. PROGRAM CONTINUED
   ```

   – Im interaktiven Dialog wird für ein Programm ohne definierte STXIT-Routine für die Ereignisklasse „Ende der Programmlaufzeit“ die Meldung `EXC0075` ausgegeben. Der Benutzer kann angeben, ob er die Ausführung fortsetzen oder beenden will. Wird die Ausführung fortgesetzt, wiederholt sich der Vorgang nach 30 CPU-Sekunden.

   ```
   EXC0075 TIME LIMIT FOR PROGRAM RUN EXCEEDED. PROGRAM TO BE CONTINUED?
           REPLY (Y=YES; N=NO)
   ```

   – In einem Programm mit einer solchen STXIT-Routine wird diese gestartet.
   Zur Abarbeitung der Routine werden der Task weitere 30 CPU-Sekunden zugestanden. Beendet die STXIT-Routine das Programm nicht, gibt sie die Kontrolle wieder an das Programm zurück. Sind die zugestandenen 30 CPU-Sekunden noch nicht aufgebraucht, läuft das Programm bis zur Überschreitung dieses neuen Zeitlimits weiter. Danach werden folgende zwei Meldungen ausgegeben:

   ```
   EXC0073 MAXIMUM PROGRAM RUNTIME EXCEEDED. 'STXIT' ROUTINE OR
           PROGRAM RUN USED UP 30 SECONDS.
   EXC0075 TIME LIMIT FOR PROGRAM RUN EXCEEDED. PROGRAM TO BE CONTINUED?
           REPLY (Y=YES; N=NO)
   ```

   Der Benutzer kann angeben, ob er die Ausführung fortsetzen oder beenden will. Wird die Ausführung fortgesetzt, wiederholt sich der Vorgang nach 30 CPU-Sekunden.

2. DIATTL=Y
   Die Dialogtask wird mit Task-Time-Runout beendet.

   Bei Überschreiten des TTL wird zunächst die Meldung `EXC0076` ausgegeben:
   ```
   EXC0076 TASK CPU TIME LIMIT for DIALOG TASK exceeded.
           Task will be terminated after 30 sec extra CPU-time.
   ```

   Falls ein Programmlauf mit STXIT für Timeout vorliegt, wird diese STXIT-Routine noch nach Ablauf der mit der Meldung eingeräumten Zeit gestartet. Der STXIT-Routine werden standardmäßig noch einmal 30 CPU-Sekunden zugestanden.

3. DIATTL=Q (quick and quiet)
   Die Dialogtask wird mit Task-Time-Runout beendet (wie bei DIATTL=Y).

   Zusätzlich werden (auch für einen Batch-Job) sowohl für TTL wie auch für PTL alle zuvor erwähnten Zeitzuschläge bzw. Wartezeiten auf eine Sekunde verkürzt („quick“) sowie die auf eine Antwort wartende Meldung `EXC0070` an der Bedienstation unterdrückt („quiet“).

## 11.4 PCS: Performance-Überwachungssystem

Das optionale Subsystem PCS (Performance Control System) unterstützt die Systembetreuung bei der optimalen Einstellung der Systeme. Es ermöglicht die Aufteilung der Leistung eines Systems auf die einzelnen Task-Kategorien und Tasks entsprechend den Erfordernissen der Benutzer.

PCS regelt bei komplexem Mischbetrieb die verschiedenen Lastanteile entsprechend den kundenspezifischen Anforderungen. PCS passt in kurzen Abständen die Stellgrößen so an, dass vorgegebene Optimierungsstrategien (Antwortzeit- bzw. Durchsatz-Optimierung) und der hierfür optimale Betriebspunkt jederzeit gehalten werden.

PCS überwacht während des Betriebes die Leistungsaufnahme der Tasks und Kategorien und die Auslastung von CPU, Hauptspeicher und Ein-/Ausgabe-System.

Wenn diese Größen nicht den vom Systembetreuer vorgegebenen Führungsgrößen (PCS-Option) entsprechen, korrigiert PCS die Leistungsaufnahme durch Veränderung BS2000-interner Stellgrößen (z.B. Task-Priorität, MIN-MPL- und MAX-MPL-Werte der Katagorien).

PCS ermittelt seine Regelgrößen über Zeiträume von etwa 10 Sekunden und passt die Stellgrößen der momentanen Lastsituation in diesem Zeitraum an.
PCS arbeitet mit PRIOR, dessen Steuerung der Task-Prioritäten beim Start von PCS abgeschaltet wird, und mit dem Job-Management zusammen.

Nährere Informationen zu PCS finden Sie im Handbuch „PCS“ [38].

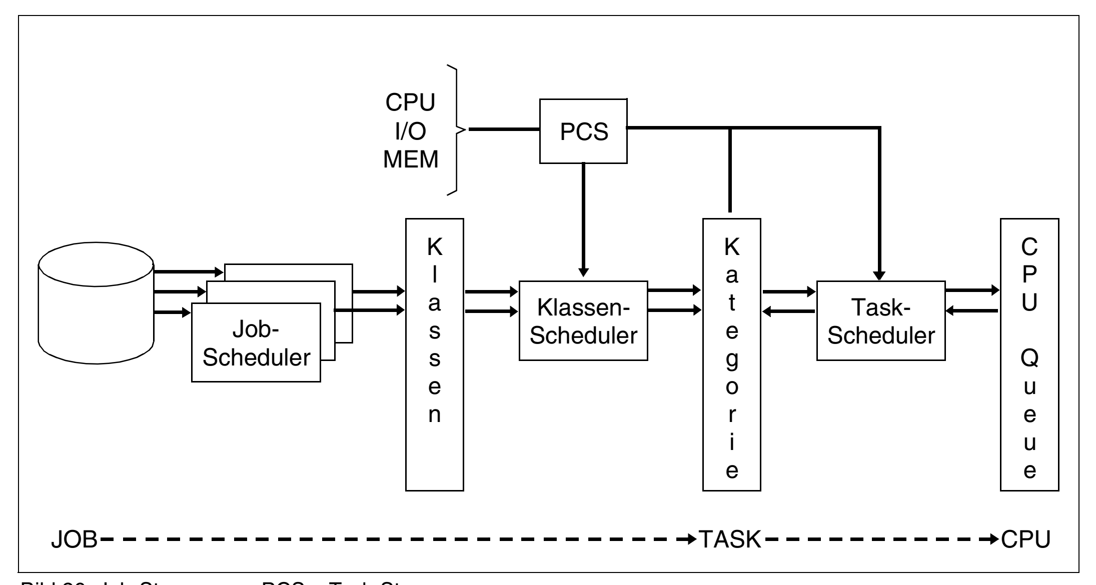



Bild 26: Job-Steuerung – PCS – Task-Steuerung

# 12 Sicherheit

Dieses Kapitel beschreibt die wesentlichen Sicherheitsmechanismen von BS2000/OSD.

Dazu gehören die Zugangs- und Zugriffsschutzmechanismen von BS2000/OSD-BC sowie die mit dem kostenpflichtigen Software-Produkt SECOS gebotenen zusätzlichen Mechanismen. Informationen zu SECOS finden Sie im Abschnitt „Erfüllung von Sicherheitsanforderungen durch SECOS“ auf Seite 528 und in den SECOS-Handbüchern [48] und [49].

## 12.1 Zugangsschutz

Zugangsschutz beinhaltet alle Methoden zum Schutz eines DV-Systems vor unberechtigtem Systemzugang.

### BS2000/OSD-BC

Der Zugangsschutz mit LOGON-Kennwort im Kommando SET-LOGON-PARAMETERS ist das derzeit gebräuchlichste Authentisierungsverfahren.
Dabei wird mit dem Kommando MODIFY-USER-PROTECTION ein 8 bzw. 32 Byte langes Kennwort für die Benutzerkennung vereinbart.

Durch organisatorische Maßnahmen, die SECOS anbietet, kann der Kennwortschutz in BS2000/OSD erheblich verbessert werden.
Dabei legt die Benutzerverwaltung für den Benutzer explizit fest, welche der folgenden Vorgaben er zwingend beachten muss:

- Minimale Länge eines Kennwortes
- Komplexität eines Kennwortes
- Lebensdauer eines Kennwortes

### Erweiterter Zugangsschutz mit SECOS

SECOS bietet folgende zusätzliche Möglichkeiten für den Zugangsschutz:

- Trennung der Zugangswege
- Einschränkung des Zugangs über Terminal-Sets
- Zugangsschutz mit Guards
- Persönliche Identifizierung
- Protokollierung der Zugangsversuche
- Sperre von Terminals oder Benutzerkennungen nach einer festgelegten Anzahl von erfolglosen Zugangsversuchen
- Sperre von Benutzerkennungen bei Inaktivität

## 12.2 Zugriffsschutz

Zugriffsschutz bezeichnet die Regeln, nach denen in einem DV-System Subjekte auf Objekte zugreifen können und die Methoden, mit denen die Einhaltung dieser Regeln sichergestellt werden kann.

**BS2000/OSD-BC**

BS2000/OSD-BC stellt Zugriffsschutzmechanismen für Objekte, die vom DVS (Dateiverwaltungssystem) verwaltet werden können, zur Verfügung. Das DVS stellt die Objektverwaltung für Dateien dar, die die Objekte des DVS sind.

Für Dateien gibt es folgende Schutzmechanismen:

- Begrenzter Pubset-Zugriff
  Durch Verteilung von Benutzerkennungen auf unterschiedliche Pubsets kann ein Schutz vor unerlaubtem Zugriff auf Objekten, z.B. Dateien, jeweils anderer Pubsets erreicht werden (siehe auch Kapitel „BS2000-Benutzerverwaltung“ auf Seite 257 und Kapitel „Pubset-Verwaltung“ auf Seite 307).

- Standard-Zugriffskontrolle (USER-ACCESS/ACCESS)
  Mit dem Zugriffsschutzmechanismus der Standard-Zugriffskontrolle (mit den Operanden ACCESS und USER-ACCESS der Kommandos CREATE-FILE und MODIFY-FILE-ATTRIBUTES) legt der Benutzer Zugriffsrechte für sich und andere fest (siehe Seite 508).

- Einfache Zugriffskontroll-Liste (Basic Access Control List, BACL)
  Mit dem Zugriffsschutzmechanismus der BACL legt der Benutzer Zugriffsrechte auf Objekte, z.B. Dateien, für eine differenzierte Menge von Subjekten fest (siehe Seite 509). Die Zugriffsrechte Lesen (read), Schreiben (write) und Ausführen (exec) können für jede der Benutzerklassen Owner, Group und Others getrennt vergeben werden.

- Kennwort
  Für jede seiner Dateien kann der Benutzer Kennwörter (Lese-, Schreib- und Ausführungs-Kennwort) vereinbaren. Vor der Verarbeitung kennwortgeschützter Dateien ist das entsprechende Kennwort anzugeben. Kennwörter können auf Wunsch verschlüsselt werden (siehe Seite 503).

- Dateiverschlüsselung
  Der Benutzer kann eine Datei in eine verschlüsselte Form umwandeln und dabei ein Crypto-Kennwort (wie ein „normales“ Kennwort, jedoch 8 Zeichen lang) vorgeben. Der Zugang zum Dateiinhalt ist dann nur noch nach Vorgabe des richtigen Crypto-Kennwort möglich (siehe Seite 504).
  Verschlüsselte Dateien haben kein Kennwort für Lesen oder Ausführen, sind aber sonst mit allen anderen Zugriffsschutzmechanismen kombinierbar.

- Schutzfrist
  Ein Benutzer kann seiner Datei eine Schutzfrist zuordnen, innerhalb der diese Datei nicht geändert werden darf (siehe Seite 503).

### openCRYPT

Die kostenpflichtigen openCRYPT-Produkte verwenden standardisierte, offene Schnittstellen zur Ver- und Entschlüsselung von Daten, siehe Handbuch „openCRYPT“ [9].

Das Subsystem openCRYPT-SERV bietet kryptografische Funktionen für S-Server. Die Verschlüsselungsaufträge können entweder vom Subsystem selbst oder wahlweise von einem kanalgekoppelten, externen Co-Prozessor ausgeführt werden. Das Ver- und Entschlüsseln von Daten im Subsystem selbst ist sehr CPU-intensiv und nur für ausgewählte, sensitive Daten sinnvoll. Das Ver- und Entschlüsseln umfangreicher Datenmengen sollte in einem externen Co-Prozessor, der openCRYPT-BOX, ausgeführt werden.

Auf SQ-Servern ist openCRYPT-SOFT als optionale, integrierte Lösung für kryptografische Funktionen verfügbar. openCRYPT-SOFT bietet eine Funktionsschnittstelle auf Basis des Public Key Cryptography Standard #11 (PKCS#11) und stellt zugleich die kryptografischen Funktionen bereit. Das CPU-intensive Ver- und Entschlüsseln von Daten wird in einer eigenen Crypto-CPU ausgeführt.

### Erweiterter Zugriffsschutz mit SECOS

Das kostenpflichtige Software-Produkt SECOS (Komponente GUARDS) stellt ein komfortables und flexibles Verwaltungsinstrument zur Verfügung, mit dem für Objekte wie Dateien, Bibliotheken, Bibliothekselemente oder weitere Objekte verschiedener Objektverwaltungen Zugriffsbedingungen definiert und bewertet werden können. Der Zugriffsschutz für die genannten Objekte wird dabei über so genannte Guards realisiert, die Zugriffsbedingungen aufnehmen. Zugriffsbedingungen werden von Objektverwaltungen (DVS, LMS, JVS, HSMS, SRPM, usw.) abgefragt. Je nach Ergebnis der Auswertung gestattet die Objektverwaltung dann dem Subjekt den Zugriff oder nicht.

### Verwendung der Schutzmechanismen

Die folgende Tabelle zeigt, welche Objekttypen mit welchem Schutzmechanismus geschützt werden können:

| Objekt | Schutzmechanismus | Begrenzter Pubset-Zugriff | ACCESS USER-ACCESS | BACL | Kennwort | Dateiverschlüsselung | Schutzfrist | GUARDS |
|---|---|---|---|---|---|---|---|---|
| **Datei** [1] | **public** | + [2] | + | + | + | + | + | + |
| | **temporär** | - [3] | - | - | - | + | - | - |
| | **privat** | - | + | + | + | - | + | - |
| | **Band** | - | + | - | + | - | + | - |
| **Dateigenerations-gruppe** | **Index public, FGen public** | - | + | + | + | + | + | + |
| | **Index public, FGen Band** | - | + | + | + | - | + | + |
| | **Index privat, FGen privat** | - | + | + | + | - | + | - |
| **Jobvariable** | **permanent** | + | + | + | + | - | + | + |
| | **temporär** | - | - | - | - | - | - | - |
| **Bibliotheks-Element** [4] | | - | - | + | - | - | - | + |
| **FITC-Port** | | - | - | - | - | - | - | + |
| **Storage-Klassen** | | - | - | - | - | - | - | + |
| **HSMS-Managementklassen** | | - | - | - | - | - | - | + |

Tabelle 36: Schutzmechanismen für Objekte

[1] Ist Datei eine Bibliothek, siehe „Besonderheiten bei Bibliothekszugriffen“ auf Seite 501

[2] + bedeutet: Schutzmechanismus anwendbar

[3] - bedeutet: Schutzmechanismus nicht anwendbar

[4] siehe „Besonderheiten bei Bibliothekszugriffen“ auf Seite 501

Wie die Tabelle zeigt, können manche Objekte mit mehreren Schutzmechanismen geschützt werden. Von den Schutzmechanismen ACCESS/USER-ACCESS, BACL und GUARDS kann jeweils nur einer für ein Objekt wirksam sein (siehe nächsten Abschnitt). Alle übrigen Schutzmechanismen gelten zusätzlich.

### Hierarchie von Datei-Schutzmechanismen

Beim gleichzeitigen Einsatz der Schutzmechanismen ACCESS/USER-ACCESS, BACL und GUARDS für dasselbe Objekt könnten widersprüchliche Situationen entstehen. Um diese zu vermeiden, gilt folgende Hierarchie:

- Wenn der Schutz eines Objektes über Guards definiert ist, dann gelten ausschließlich die in den Guards festgelegten Zugriffsbedingungen. Eine ggf. für das Objekt definierte BACL und die Schutzattribute ACCESS/USER-ACCESS bleiben unberücksichtigt.
- Wenn kein Guardschutz für ein Objekt definiert ist, aber eine BACL, dann gilt die in der BACL festgelegte Schutzeinstellung. Die Schutzattribute ACCESS und USER-ACCESS bleiben unberücksichtigt.
- Wenn der Schutz eines Objektes weder mit Guards noch mit BACL geregelt ist, dann werden für den Schutzmechanismus die Schutzattribute ACCESS und USER-ACCESS herangezogen.

In allen Fällen gelten zusätzlich der begrenzte Pubset-Zugriff, der Kennwortschutz, die Dateiverschlüsselung und die Schutzfrist.

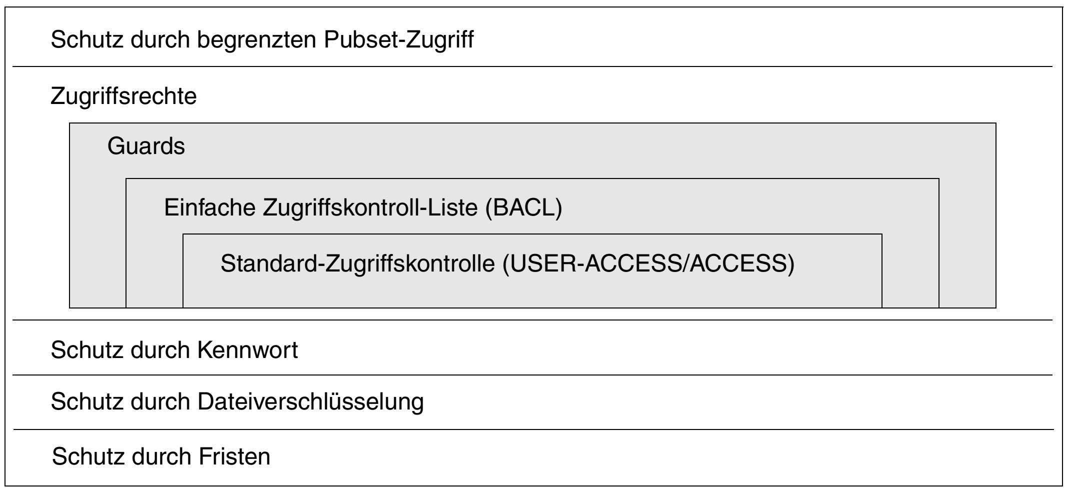



Bild 27: Schutzmechanismen für Dateien auf einem Pubset

Jede Datei und jede Jobvariable kann durch einen oder mehrere Schutzmechanismen gesichert werden. Die Zugriffsrechte sind mit folgender Priorität wirksam:

1. Guards
2. BACL
3. Standard-Zugriffskontrolle

Von den drei Schutzmechanismen wird genau einer bei jeder Überprüfung einer Zugriffsberechtigung herangezogen. Das ist derjenige, der in der Hierarchie am höchsten steht **und** der aktiviert ist.

### Besonderheiten bei Bibliothekszugriffen

PLAM-Bibliotheks**dateien** können als Ganzes wie eine Datei geschützt werden. Davon unabhängig können Bibliotheks**elemente** mit der LMS-Anweisung MODIFY-ELEMENT-PROTECTION geschützt werden.

Beim Zugriff auf Bibliotheken und Bibliothekselemente ist daher Folgendes zu beachten:

- Der Zugriff auf einzelne Bibliothekselemente wird durch die mit MODIFY-ELEMENT-PROTECTION festgelegten Schutzmechanismen geregelt. Unabhängig vom Elementschutz ist dieser Zugriff jedoch nur möglich, wenn die Bibliotheksdatei in ihrer Gesamtheit gelesen werden darf.
- Beim Zugriff auf eine Bibliothek in ihrer Gesamtheit (mit ARCHIVE, mit File-Transfer oder mit dem DVS-Kommando COPY-FILE) gilt Folgendes:

  a) Ist die Bibliothek weder durch eine BACL noch durch Guards geschützt, kann auf sie zugegriffen werden wie auf eine beliebige Datei.

  b) Für den Zugriff auf eine Bibliothek, die durch eine BACL oder ein Guard geschützt ist, können die Zugriffsregelungen folgender Tabelle entnommen werden:

| | | Bibliothek enthält mindestens ein Element, das mit einer BACL oder einem Guard geschützt ist | Bibliothek enthält **kein** Element, das mit einer BACL oder einem Guard geschützt ist |
|---|---|---|---|
| **Zugriff durch** | **Eigentümer** | * [1] | * |
| | **Miteigentümer** | * | * |
| | **Andere** | Zugriff verboten | * |

Tabelle 37: Zugriffsregelungen beim Zugriff auf Bibliotheken

[1] * bedeutet: Zugriff hängt von den Zugriffsregelungen der gesamten Bibliothek ab

## 12.3 Zugriffsschutz in BS2000/OSD

| Kommando | Bedeutung |
|---|---|
| ADD-CRYPTO-PASSWORD | Crypto-Kennwort zur Entschlüsselung von verschlüsselten Dateiinhalten wird in der Kennwort-Tabelle der Task hinterlegt |
| ADD-PASSWORD | Tasklokale Eingabe eines Kennworts in die Kennwort-Tabelle, um Zugriff zu kennwortgeschützten Dateien zu ermöglichen |
| CREATE-FILE | Name und Merkmale für eine neue Datei vereinbaren |
| CREATE-FILE-GROUP | Name und Merkmale für eine neue Dateigenerationsgruppe vereinbaren |
| CREATE-JV [1] | Neue Jobvariable erzeugen |
| DECRYPT-FILE [2] | Verschlüsselte Datei in unverschlüsselte Datei umwandeln |
| ENCRYPT-FILE [2] | Unverschlüsselte Datei in verschlüsselte Datei umwandeln und Crypto-Kennworts vereinbaren |
| MODIFY-FILE-ATTRIBUTES | Festlegen der Schutzmerkmale einer Datei |
| MODIFY-FILE-GROUP-ATTRIBUTES | Festlegen der Schutzmerkmale einer Dateigenerationsgruppe |
| MODIFY-JV-ATTRIBUTES [1] | Festlegen der Schutzmerkmale einer Jobvariablen |
| REMOVE-CRYPTO-PASSWORD | Crypto-Kennwort wird aus der Kennwort-Tabelle der laufenden Task gelöscht |
| REMOVE-PASSWORD | Kennwort aus der Kennwort-Tabelle löschen |
| SHOW-FILE-ATTRIBUTES | Information über die Schutzmerkmale einer Datei |
| SHOW-JV-ATTRIBUTES [1] | Information über die Schutzmerkmale einer Jobvariablen |

Tabelle 38: Kommandoübersicht für die Zugriffskontrolle in BS2000/OSD

[1] Diese Kommandos setzen das kostenpflichtige Software-Produkt JV voraus

[2] Diese Kommandos setzen kostenpflichtige openCRYPT-Produkte voraus, siehe Seite 498

## 12.3.1 Kennwörter und Schutzfristen

Bei der Erzeugung oder Bearbeitung einer Datei mit dem Kommando CREATE-FILE bzw. MODIFY-FILE-ATTRIBUTES können für jede Zugriffsart (READ, WRITE, EXEC) Kennwörter vergeben werden (Operand `PROTECTION=*PAR(<zugriffsart>-PASSWORD=...)`).

| | |
|---|---|
| `READ-PASSWORD=` | Anzugeben für Lesen (`*NONE`: kein Kennwort erforderlich) |
| `WRITE-PASSWORD=` | Anzugeben für Schreiben (`*NONE`: kein Kennwort erforderlich) |
| `EXEC-PASSWORD=` | Anzugeben für Ausführen (`*NONE`: kein Kennwort erforderlich) |

Bei der Änderung der Dateiattribute mit dem Kommando MODIFY-FILE-ATTRIBUTES sind die zuvor vergebenen Kennwörter gemäß der geltenden Kennworthierarchie (WRITE-PASSWORD – READ-PASSWORD – EXEC-PASSWORD) anzugeben.

Bei der Bearbeitung einer Datei (Kommandos ADD-FILE-LINK EXPIRATION-DATE= ... oder MODIFY-FILE-ATTRIBUTES ...,PROTECTION=*PARAMETERS(EXPIRATION-DATE=...) kann eine Schutzfrist vergeben bzw. geändert werden.

Für die Zeitdauer dieser Schutzfrist kann ein möglicher Schreibzugriff (durch das Attribut „w“ in der einfachen Zugriffskontroll-Liste einer Datei oder ein auf Grund von ACCESS= *WRITE bei Standardzugriffskontrolle) unwirksam gemacht werden. Die Datei kann dann nur gelesen werden.



Wenn eine mit einem Lese- und/oder Ausführungskennwort geschützte Datei umgewandelt wird, verliert die Datei dabei den Schutz durch Lese- und/oder Ausführungskennwort.

## 12.3.2 Dateiverschlüsselung

Durch die Dateiverschlüsselung mit Crypto-Kennwort ist es möglich, den Inhalt einer Datei vor unbefugtem Zugriff zu schützen – auch gegenüber Personen mit TSOS-Privileg und auch gegen den physikalischen Zugriff auf Platten und Sicherungsbänder.

Eine verschlüsselte Datei wird durch Umwandlung einer normalen Datei mit dem Kommando ENCRYPT-FILE erzeugt. Dabei erfolgt die Festlegung des Crypto-Kennworts und das im Systemparameter FILECRYP eingestellte Verschlüsselungsverfahren wird in den Katalogeintrag der Datei übernommen.

Das Kommando DECRYPT-FILE macht aus verschlüsselten Dateien wieder unverschlüsselte. Dies ist nur nach Vorgabe des korrekten Cypto-Kennworts möglich.

### Crypto-Kennwort

Ein Crypto-Kennwort ist max. 8 Zeichen lang, ohne Unterscheidung von Groß- und Kleinschreibung.

Zur Vorgabe der Zugriffsberechtigung auf eine verschlüsselte Datei wird das zugehörige Crypto-Kennwort mit dem Kommando ADD-CRYPTO-PASSWORD in der tasklokalen Crypto-Kennworttabelle hinterlegt.

Das explizite Entfernen eines Crypto-Kennworts aus der Kennworttabelle erfolgt mit dem Kommando REMOVE-CRYPTO-PASSWORD (sonst implizit beim Beenden der Session).



Ohne Angabe des korrekten Crypto-Kennworts ist eine Entschlüsselung unmöglich. Gegen Verlust des Crypto-Kennworts werden folgende Maßnahmen empfohlen:

- Hinterlegen der festgelegten Crypto-Kennworte an einem sicheren Ort
- Anzahl der verwendeten Crypto-Kennwörter klein halten. Dazu gehört:
  - Zusammengehörige verschlüsselte Dateien immer mit dem gleichen Crypto-Kennwort schützen (Benutzer: ENCRYPT-FILE mit Angabe einer Referenzdatei).
  - Neben bestimmten verschlüsselten Musterdateien nur verschlüsselte Dateien mit gleichen Crypto-Kennworten wie diese Musterdateien erlauben (Systembetreuer: Systemparameter FREFCRYP, siehe Seite 506).

### Arbeiten mit verschlüsselten Dateien

Beim **homogenen Transfer** einer verschlüsselten Datei wird der verschlüsselte Inhalt eins-zu-eins in eine Empfangsdatei übertragen, die dieselben Verschlüsselungsattribute hat wie die Sendedatei.

Dieser homogene Transfer wird benutzt bei:

- homogenem COPY-FILE
- Sichern und Restaurieren (SAVE/RESTORE) mit HSMS/ARCHIVE
- Verdrängen und Zurückholen (MIGRATION/RECALL) mit HSMS
- Export und Import mit HSMS/ARCHIVE
- Verlagern von Dateien innerhalb eines SM-Pubsets auf einen anderen Volume-Set
- homogener File-Transfer

Beim homogenen Transfer findet also keine Entschlüsselung statt und somit wird auch kein Key und kein Crypto-Kennwort benötigt. Dadurch kann insbesondere der Systembetreuer bei Transfer-Aktionen mit verschlüsselten Dateien arbeiten wie mit unverschlüsselten Dateien.

Bei der Ausgabe des Kommandos SHOW-FILE-ATTRIBUTES zeigt das Feld ENCRYPTION im Abschnitt SECURITY an, ob und mit welchem Verschlüsselungsverfahren (AES oder DES) eine Datei verschlüsselt ist. Eine Dateiselektion wird entsprechend der Werte des Operanden ENCRYPTION angeboten.

Bei Fern-Dateizugriff über RFA werden die Kommandos ADD- und REMOVE-CRYPTO-PASSWORD von der Aufrufertask automatisch an alle angeschlossenen RFA-Partnerprozesse weitergeleitet.

DAB unterstützt das Lese-Caching von verschlüsselten und unverschlüsselten Dateien, nicht jedoch das Schreib-Caching, siehe Seite 303.
Lese-Caching ist für verschlüsselte Dateien günstig, um die durch das Verschlüsselungsverfahren verlängerten Zugriffszeiten zu verkürzen.

### Einschränkungen und Besonderheiten

- Durch die Bindung an das Subsystem openCrypt ist ab „System Ready“ der Zugriff auf verschlüsselte Dateien möglich.
- Nicht verschlüsselt werden Jobvariablen, Banddateien, EAM-Dateien, Dateien auf Privatplatte und TSOS-Dateien auf dem Home Pubset.
- Verschlüsselte Dateien können nicht gedruckt werden. Eine verschlüsselte Datei muss erst entschlüsselt werden.

- Verschlüsselte Dateien haben kein Kennwort für Lesen und Ausführen. Sie können jedoch ein Schreib-Kennwort haben und sind auch mit den anderen Zugriffsschutzmechanismen kombinierbar.
- Die Änderung des Crypto-Kennworts mit dem Kommando MODIFY-FILE-ATTRIBUTES ist nicht möglich. Änderungen von READ- oder EXEC-PASSWORD werden ignoriert.
- Bei Einsatz von PAMCONV oder bei Aufruf des Kommandos REPAIR-DISK-FILE für eine verschlüsselte Datei ist die Vorgabe des Crypto-Kennworts erforderlich.

**Möglichkeiten der Steuerung über Systemparameter**

1. FILECRYP

   Der Systemparameter FILECRYP bestimmt das Verschlüsselungsverfahren beim Umwandeln in eine verschlüsselter Datei mit dem Kommando ENCRYPT-FILE.
   Es werden die Verschlüsselungsverfahren AES (Voreinstellung) und DES unterstützt.

   – Der aktuelle Wert des Systemparameters FILECRYP wird bei ENCRYPT-FILE in die Verschlüsselungsmerkmale der Datei übernommen.

   – Beim Zugriff auf eine bereits verschlüsselte Datei wird das Verschlüsselungsverfahren nicht dem Systemparameter FILECRYP entnommen, sondern dem Katalogeintrag der Datei.

   – Eine Änderung des Systemparameters FILECRYP wird für eine zum Zeitpunkt der Änderung bereits verschlüsselte Datei erst dann wirksam, wenn die Datei entschlüsselt und anschließend erneut verschlüsselt wird.

   Beim Shared-Pubset-Betrieb mit verschlüsselten Dateien sollte die Wahl der einzelnen Verschlüsselungsverfahren im Systemparameter FILECRYP der beteiligten Systeme einheitlich sein.

2. FREFCRYP

   Zur kontrollierten Vergabe und zahlenmäßigen Begrenzung der verwendeten Crypto-Kennwörter steht der Systemparameter FREFCRYP zur Verfügung.
   Enthält - falls nicht leer - eine ausgewählte Benutzerkennung. Nur für Dateien von dieser Benutzerkennung ist dann die Umwandlung in verschlüsselte Dateien (Kommando ENCRYPT-FILE) mit freier Definition eines Crypto-Kennworts möglich.
   Bei der Umwandlung von Dateien anderer Benutzerkennungen muss dann eine schon verschlüsselte Referenzdatei angegeben werden. Die Menge aller verwendeten Crypto-Kennwörter wird damit begrenzt auf die der verschlüsselten Dateien von der ausgewählten Benutzerkennung.

3. PWACTIVE

   Mit dem Systemparameter PWACTIVE wird festgelegt, wie viele Crypto-Kennwörter eine Crypto-Kennworttabelle maximal enthalten darf. Wird der Grenzwert erreicht, wird die Meldung `DMS0691` ausgegeben und es kann für die laufende Task kein weiteres Crypto-Kennwort mehr eingegeben werden, bevor nicht mindestens eines der Crypto-Kennwörter aus der Crypto-Kennworttabelle wieder ausgetragen wurde.

4. PWENTERD

   Mit dem Systemparameter PWENTERD wird festgelegt, wie viele Crypto-Kennwörter pro Task eingegeben werden dürfen. Wird der Grenzwert erreicht, wird die Meldung `DMS0692` ausgegeben und es kann für die laufende Task kein weiteres Crypto-Kennwort mehr eingegeben werden.

5. PWERRORS

   Mit dem Systemparameter PWERRORS wird festgelegt, wie viele Crypto-Fehlzugriffsversuche unter einer Task maximal toleriert werden. Wird der Grenzwert erreicht, wird ein SAT-Eintrag geschrieben bzw. die Meldung `DMS0693` auf die Konsole ausgegeben, wenn SECOS nicht installiert ist.

6. PWPENTI

   Mit dem Systemparameter PWPENTI wird eine Festlegung für die Zeitstrafe der zu tolerierenden Crypto-Fehlzugriffsversuche getroffen.



Alle oben aufgeführten Grenzwerte gelten sowohl für die Eingabe von Crypto-Kennwörtern als auch für die Eingabe von Datei-Kennwörtern (READ, WRITE, EXEC). Im Gegensatz zu den Datei-Kennwörtern, bei deren Eingabe ein privilegierter Systembetreuer das Sonderrecht besitzt, die festgelegten Grenzwerte zu überschreiten, werden bei der Eingabe von Crypto-Kennwörtern keinerlei Sonderrechte gewährt.

Crypto-Kennworttabellen und Crypto-Kennwortzähler werden getrennt von den Datei-Kennworttabellen und den Datei-Kennwortzählern geführt. Ist zum Beispiel unter einer Task die maximal mögliche Anzahl von READ-, WRITE- und EXEC-Kennworteingaben erreicht, sind durchaus weitere Crypto-Kennworteingaben möglich und umgekehrt.

Der Parameter ENCRYPT wirkt sich nur aus auf die Verschlüsselung von Dateikennworten, nicht auf die von Crypto-Kennworten. Diese werden stets mit Einweg-Verschlüsselung in der Crypto-Kennworttabelle hinterlegt.

Für weitere Hinweise zur Nutzung von verschlüsselten Dateien siehe Handbuch „Einführung in das DVS“ [19].

### 12.3.3 Standard-Zugriffskontrolle (ACCESS/USER-ACCESS)

Gesteuert wird die Standard-Zugriffskontrolle über die Operanden ACCESS und USER-ACCESS der CREATE- und MODIFY-Kommandos.

Die Standard-Zugriffskontrolle über die Schutzattribute ACCESS und USER-ACCESS bietet sich dann an, wenn ein höherer Zugriffsschutz durch die einfache Zugriffskontroll-Liste nicht gewünscht wird.

Ist weder eine BACL noch ein Guard definiert, wird die Standard-Zugriffskontrolle wirksam. Zusätzlich werden immer die Kennwörter und die Schutzfrist überprüft.

#### Schutzattribut **ACCESS**

Mit dem Schutzattribut ACCESS kann für ein Objekt Schreib- oder Leserecht festgelegt werden. Das Schreibrecht schließt hierbei das Leserecht ein.

#### Schutzattribut **USER-ACCESS**

Mit dem Schutzattribut USER-ACCESS kann festgelegt werden, ob auf eine Datei nur der Eigentümer (*USER-ONLY) oder auch alle übrigen Benutzer (*ALL-USERS) – ausschließlich oder einschließlich der Benutzerkennung(en) mit dem Privileg HARDWARE-MAINTENANCE für die Online-Wartung) – zugreifen dürfen.

*Anmerkung für Benutzerkennungen mit dem Privileg HARDWARE-MAINTENANCE*

Ein Dateizugriff wird einer Benutzerkennung mit diesem Privileg nur dann erlaubt, wenn folgende Bedingungen erfüllt sind:

- Falls die Datei mit Guards geschützt ist, müssen im Guard Zugriffsbedingungen festgelegt sein, die der privilegierten Benutzerkennung den Zugriff erlauben.
- Falls die Datei nicht mit Guards geschützt ist, aber durch eine einfache Zugriffskontroll-Liste (BACL), muss diese der privilegierten Benutzerkennung den Zugriff erlauben.
- Wenn die Datei weder mit Guards noch mit einer BACL geschützt ist, muss USER-ACCESS=*SPECIAL gesetzt sein.

## 12.3.4 Einfache Zugriffskontroll-Liste (BACL)

Eine Stufe über den Schutzattributen ACCESS/USER-ACCESS liegt in der Hierarchie der Zugriffsschutzmechanismen die einfache Zugriffskontroll-Liste (Basic Access Control List, BACL). Sie ist für ein Objekt dann wirksam, wenn für das Objekt kein Guards-Schutz definiert ist. Kennwortschutz und Schutzfrist sind zusätzlich wirksam.

Mit einer BACL können für den Objekt-Eigentümer, für die Mitglieder seiner Benutzergruppe und für alle anderen Benutzer unterschiedliche Zugriffsrechte definiert werden. Eine Festlegung von Zugriffsrechten für einzelne Benutzerkennungen ist mit diesem Zugriffsschutzmechanismus jedoch nicht möglich.

Eine einfache Zugriffskontroll-Liste für Dateien wird mit dem Operanden BASIC-ACL der Kommandos CREATE-FILE oder MODIFY-FILE-ATTRIBUTES definiert.

Einfache Zugriffskontroll-Listen für Jobvariablen können entsprechend mit den Kommandos CREATE-JV oder MODIFY-JV-ATTRIBUTES definiert werden.

*Benutzerklassen*

Aufbauend auf dem Konzept der Benutzergruppen werden Benutzerklassen für den Zugriff auf Objekte festgelegt. Die Benutzerklassen unterteilen die Menge aller Benutzer jeweils in die Teilmengen OWNER, GROUP und OTHERS.

OWNER der Eigentümer eines Objekts, also die Benutzerkennung, unter der die Datei oder Jobvariable katalogisiert ist, sowie Miteigentümer, die mit Hilfe des Miteigentümerschutzes festgelegt wurden

GROUP alle Benutzerkennungen der Benutzergruppe, der der Eigentümer angehört, mit Ausnahme des Eigentümers selbst und der Miteigentümer

OTHERS alle übrigen Benutzer mit Ausnahme der Miteigentümer

Zur Festlegung der Benutzerklasse wird die Definition der Gruppenstruktur auf dem Home-Pubset herangezogen.

*Hinweise zur Benutzerklasse GROUP*

Alle Benutzer, die keiner explizit eingerichteten Gruppe zugeordnet sind, sind automatisch Mitglied der implizit eingerichteten Gruppe *UNIVERSAL. Dies gilt insbesondere dann, wenn gar keine Gruppen explizit eingerichtet wurden. In diesem Fall sind alle Systembenutzer Mitglied derselben Gruppe. Bei der Auswertung einer BACL erhalten daher alle zugreifenden Benutzerkennungen außer dem Objekteigentümer selbst die Zugriffsrechte aus dem GROUP-Eintrag und nicht die des OTHERS-Eintrags.

Für Mitglieder der Benutzergruppe *UNIVERSAL wird daher dringend empfohlen, für die Benutzerklassen GROUP und OTHERS die gleichen Zugriffsrechte zu vergeben.

*Zugriffsrechte*

In einer BACL sind neun Zugriffsberechtigungen für eine Datei festgelegt. Der Datei können für jede der drei Benutzerklassen OWNER, GROUP, OTHERS drei Zugriffstypen separat zugeordnet werden:

- Lesen (R)
- Schreiben (W)
- Ausführen (X)

Keines dieser Zugriffsrechte schließt ein anderes ein.

*Auswertung der einfachen Zugriffskontroll-Liste*

1. Ist die Benutzerkennung, die den Zugriff wünscht, der Eigentümer oder ein Miteigentümer des Objekts bzw. TSOS, gelten die unter OWNER abgespeicherten Zugriffsrechte.
2. Gehört die Benutzerkennung der Benutzergruppe des Eigentümers an, gelten die unter GROUP abgespeicherten Zugriffsrechte.
3. Für alle anderen Benutzerkennungen gelten die unter OTHERS abgespeicherten Zugriffsrechte.

*Beispiel*

| OWNER | | | GROUP | | | OTHERS | | |
|---|---|---|---|---|---|---|---|---|
| R | W | X | R | W | - | R | - | - |

Der Eigentümer dieser Datei darf auf die Datei lesend, schreibend und ausführend zugreifen. Die Gruppe des Dateieigentümers darf die Datei lesen und in die Datei schreiben. Der Rest darf die Datei nur lesen.

# 12.4 Privilegien

Systemprivilegien (kurz: Privilegien) beschreiben das Recht, bestimmte Aufgabenbereiche der Systembetreuung mit den dazu erforderlichen Systemfunktionen unter einer bestimmten Benutzerkennung abzuwickeln.

Privilegien können an unterschiedliche Benutzerkennungen vergeben werden. Diese Verteilung bewirkt zum einen eine Entlastung der Systembetreuung. Zum anderen wird dadurch die Sicherheit im Rahmen der Systembetreuung erhöht, weil z.B. der Kreis derer kleiner wird, die für anfallende Routinearbeiten das TSOS-Kennwort kennen müssen.

Standardmäßig sind die Privilegien an vordefinierte Benutzerkennungen vergeben, siehe Tabelle 40 auf Seite 524. Die Standard-Privilegienverteilung kann nur mit SECOS geändert werden.

Jedes Kommando muss in einer der aktivierten System-Syntaxdateien deklariert und seine Verwendung ausdrücklich zugelassen sein. Jedes an SDF übergebene Kommando (Benutzer-, Systemverwaltungs- oder Operatorkommando) wird vor der Verarbeitung daraufhin überprüft, ob der Aufrufer mit dem zur Ausführung nötigen Privileg ausgestattet ist, im Falle der Operatorkommandos ist dies z.B. das Privileg OPERATING.

Die folgende Übersicht zeigt anhand einiger Kommandos die Aufteilung der Berechtigungen, Benutzer-, Operator- und Systemverwaltungskommandos an BS2000/OSD zu geben:

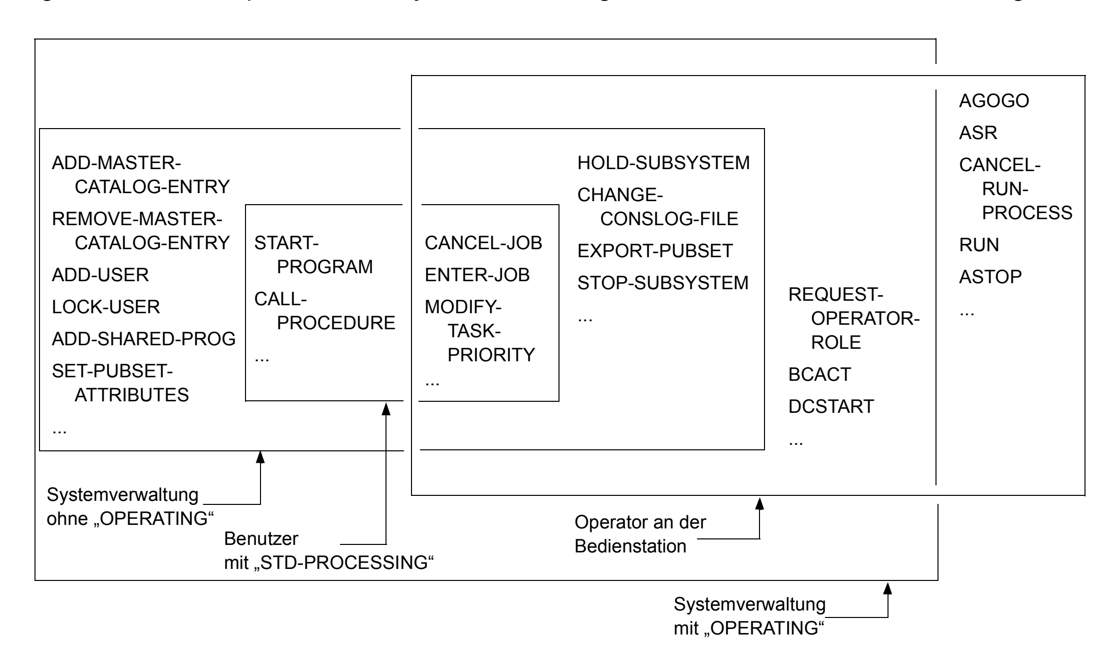



Bild 28: Berechtigungen zur Eingabe von Kommandos

Es gibt also einige Kommandos, die sowohl vom Operator (über Bedienstation) als auch von der Systemverwaltung (unter der Benutzerkennung TSOS) eingegeben werden können. Die Aufgabenteilung zwischen Systemverwaltung und Operator ist deshalb nicht starr. Hier ist ein gewisser Spielraum in der Organisation des Data Centers vorgesehen. In jedem Fall muss die Zusammenarbeit zwischen Systemverwaltung und Operator sehr eng sein.

### Privilegienverteilung mit SECOS

Jede Benutzerkennung in BS2000/OSD ist mit mindestens einem Privileg ausgestattet. Standardmäßig ist das das Privileg STD-PROCESSING, d.h. das Recht, die Benutzerkommandos auszuführen.

Eine Benutzerkennung kann mehrere Privilegien besitzen und ein Privileg kann mit SECOS an mehrere Benutzerkennungen vergeben werden.

Mit SECOS können einzelne Privilegien zu bestimmten Aufgabenbereichen gruppiert werden. Eine solche Gruppierung wird durch die Definition von „Sammelprivilegien“ realisiert, denen die verschiedenen (Einzel-)Privilegien zugeordnet werden können.

Die Privilegien einer Benutzerkennung werden im Benutzerkatalog (SYSSRPM) hinterlegt. Die Privilegienverteilung im Benutzerkatalog des Home-Pubsets ist systemweit wirksam. Der Benutzerkatalog eines Pubsets wird beim Importieren dieses Pubsets geöffnet und bleibt offen bis zum Exportieren des Pubsets.

Die einzelnen Privilegien sind im Abschnitt „Beschreibung der Privilegien“ auf Seite 516 beschrieben.

## 12.4.1 Privilegien der Benutzerkennung TSOS

Das Privileg **TSOS** ist fest an die Benutzerkennung TSOS geknüpft und kann weder dieser Benutzerkennung entzogen noch an eine andere Benutzerkennung vergeben werden. Welche Privilegien der Benutzerkennung TSOS standardmäßig zugewiesen sind, kann der Tabelle 40 auf Seite 524 entnommen werden.



Gemäß BS2000-Konvention beginnen Produktdateien oder vom System unter der Benutzerkennung TSOS angelegte Dateien mit der Zeichenfolge „SYS“ (für S-Server) oder „SKM“ und „SKU“ (für SQ-Server). Es muss darauf geachtet werden, dass sie nicht versehentlich überschrieben oder gelöscht werden.

### Benutzerkommandos

Die Benutzerkennung TSOS ist Miteigentümer aller Dateien und Jobvariablen sämtlicher Benutzer, solange ihr diese Miteigentümerschaft nicht durch den Eigentümer selbst entzogen wird.

Der Systembetreuung steht der volle Umfang der Benutzerkommandos zur Verfügung.

TSOS ist berechtigt, auf alle Einträge im Datei- oder Benutzerkatalog lesend und schreibend zuzugreifen. Der Kennwortschutz von Benutzerdateien ist für die Systembetreuung insofern aufgehoben, als TSOS berechtigt ist, entweder die Kennwörter zu ignorieren oder bei Bedarf zu ermitteln.
Mit dem Einsatz von SECOS kann der Eigentümer seine sicherheitskritischen Dateien mit einem Bedingungs-Guard versehen, der TSOS den Zugriff verbietet.

Der Zugang zu allen Dateien schließt auch temporäre Dateien ein, die die Systembetreuung unter jeder beliebigen Katalog- und Benutzerkennung einrichten darf. Diese Dateien werden jedoch **nicht** automatisch bei EXIT-JOB bzw. LOGOFF gelöscht. Für das Löschen dieser Dateien ist die Systembetreuung selbst verantwortlich. Temporäre Dateien werden bei einer teilqualifizierten Angabe von Dateinamen entsprechend ihrer internen Darstellung berücksichtigt.

### Kommandos mit anderen Privilegien als TSOS

Die Benutzerkennung TSOS kann nur die Kommandos ausführen, für die sie auch das entsprechende Privileg besitzt.

Insbesondere können unter TSOS keine Operatorkommandos eingegeben werden, da TSOS standardmäßig nicht das Privileg OPERATING besitzt.
Die Standard-Privilegienverteilung kann nur mit SECOS geändert werden.

### Makros

Die Privilegien für das Einrichten und Verändern von Dateien auf Kommandoebene gelten ebenfalls bei den entsprechenden Makroaufrufen.
Die ausführliche Beschreibung der privilegierten Operanden der Makros ist den Handbüchern „Makroaufrufe an den Ablaufteil“ [30] und „DVS Makros“ [20] zu entnehmen.

### Jobvariablen

Die nachfolgenden Kommandos stehen nur im Zusammenhang mit dem Software-Produkt JV zur Verfügung und sind im Handbuch „Jobvariablen“ [26] ausführlich beschrieben.

Die Benutzerkennung TSOS ist Miteigentümer aller Dateien und Jobvariablen sämtlicher Benutzer, solange ihr diese Miteigentümerschaft nicht durch den Eigentümer selbst entzogen wird.

Die Systembetreuung kann mit den folgenden Kommandos Jobvariablen anderer Benutzerkennungen bearbeiten:

```
CREATE-JV
DELETE-JV
MODIFY-JV
MODIFY-JV-ATTRIBUTES
MODIFY-JV-CONDITIONALLY
REMOVE-JV-LINK
SET-JV-LINK
SHOW-JV
SHOW-JV-ATTRIBUTES
SHOW-JV-LINK
SHOW-CJC-STATUS
```

Eventuell vereinbarter Dateischutz in Form von Kennwörtern wird nur dem privilegierten Aufrufer unter der Benutzerkennung TSOS angezeigt.

### 12.4.2 Privilegien für das Operating

Das Privileg **OPERATING** ist standardmäßig der Benutzerkennung SYSOPR zugeordnet. Die Standard-Privilegienverteilung kann nur mit SECOS geändert werden.

Damit ist es möglich, Operating sowohl von physikalischen und logischen Konsolen als auch von Benutzertasks mit dem Privileg OPERATING zu betreiben.
Für Konsolen bleibt das Kommandoberechtigungskonzept über Berechtigungsschlüssel unverändert erhalten. Für Benutzertasks spielt es keine Rolle mehr (Ausnahme: Kommandos, die über die UCON-Task abgearbeitet werden und denen der Berechtigungsschlüssel $ zugeordnet ist, siehe auch Seite 686).

Der Systembetreuung ist es freigestellt, bestimmte Kommandos, die der Operator grundsätzlich verwenden dürfte, in der Systemsyntaxdatei zu sperren (siehe auch Handbuch „SDF-A“ [46]).
Eine Task mit dem Privileg OPERATING bietet die meisten Operator-Kommandos an, also auch diejenigen, welche bisher ausschließlich über Bedienstation eingegeben werden konnten. Nicht angeboten werden z.B. die Kommandos CONSOLE, ASR, RUN, AGOGO.

Jedes von einer Bedienstation eingegebene Operatorkommando wird durch eine OPR-Task verwaltet, die sich beim Ausführen eines Kommandos gegenüber SDF und dem System als privilegiert ausweist. Jede Anforderung des Operators (von der OPR-Task zur Syntax- und Privilegienprüfung an SDF übergebene Kommandos) wird vor der Verarbeitung daraufhin überprüft, ob der Aufrufer mit dem zur Ausführung nötigen Privileg OPERATING ausgestattet ist.
Die OPR-Task einer physikalischen Konsole (bei NBCONOPI=NO) und einer logischen Konsole besitzt immer das Privileg OPERATING.

Der Operator kann die Kommandos über die Bedienstation eingeben.
Die Kurzform ist, falls vorgesehen, in der Kommandosyntax angegeben.
Zum Aufbau der Kommandoeingaben und Meldungen siehe Kapitel „Operatorfunktionen“ auf Seite 605 und Kapitel „Automatisierung von Operatorfunktionen“ auf Seite 663.

Die Kommandos sind im Handbuch „Kommandos“ [27] beschrieben. Die BCAM-Kommandos für den Operator (z.B. BCACT, DCSTART) sind im Handbuch „BCAM“ [4] beschrieben.

Unter bestimmten Voraussetzungen kann der Operator auch einige Systemverwaltungs- bzw. Benutzerkommandos verwenden (siehe Abschnitt „Ausübung von Systemverwaltungsfunktionen durch den Operator“ auf Seite 702).

## 12.4.3 Beschreibung der Privilegien

BS2000/OSD kennt folgende Privilegien.
Sie sind im Anschluss an diese Tabelle beschrieben.

| Verwaltungsbereich | Name des Privilegs |
|---|---|
| ACS-Verwaltung | ACS-ADMINISTRATION |
| (ist nicht vordefiniert,<br>wird vom Systembetreuer bestimmt) | CUSTOMER-PRIVILEGE-1<br>:<br>CUSTOMER-PRIVILEGE-8 |
| File-Transfer-Verwaltung | FT-ADMINISTRATION |
| FTAC-Verwaltung | FTAC-ADMINISTRATION |
| Systemglobale Guard-Administration | GUARD-ADMINISTRATION |
| Hardware-Online-Wartung | HARDWARE-MAINTENANCE |
| HSMS-Verwaltung | HSMS-ADMINISTRATION |
| Netzverwaltung (obsolet) | NET-ADMINISTRATION |
| Notification-Service-Administration | NOTIFICATION-ADMINISTRATION |
| Operating | OPERATING |
| POSIX-Benutzerverwaltung | POSIX-ADMINISTRATION |
| SPOOL-Verwaltung | PRINT-SERVICE-ADMINISTRATION |
| PROP-XT-Verwaltung | PROP-ADMINISTRATION |
| Audit-Datei-Auswertung | SAT-FILE-EVALUATION |
| Audit-Datei-Verwaltung | SAT-FILE-MANAGEMENT |
| Sicherheitsverwaltung | SECURITY-ADMINISTRATION |
| Ausführen von Benutzerkommandos | STD-PROCESSING |
| Subsystem-Verwaltung | SUBSYSTEM-MANAGEMENT |
| Software-Monitor-Verwaltung | SW-MONITOR-ADMINISTRATION |
| Bandverwaltung | TAPE-ADMINISTRATION |
| Encryption-Key-Verwaltung für Bänder | TAPE-KEY-ADMINISTRATION |
| TSOS | TSOS |
| Benutzerverwaltung | USER-ADMINISTRATION |
| VM-Verwaltung | VIRTUAL-MACHINE-ADMINISTRATION |
| VM2000-Verwaltung | VM2000-ADMINISTRATION |

Tabelle 39: Verwaltungsbereiche und dazugehörende Privilegien

### ACS-Verwaltung

Die mit dem Privileg **ACS-ADMINISTRATION** ausgestattete Benutzerkennung darf im Rahmen der Funktion ACS (Alias Catalog Service) folgende Rechte wahrnehmen:

- systemglobale Voreinstellungen und Einschränkungen für die Benutzung eines Aliaskataloges festlegen
- die Vereinbarungen hinsichtlich der ACS-Systemdateien treffen bzw. ändern
- erweiterte Funktionen einzelner ACS-Kommandos wahrnehmen

Das Privileg ist standardmäßig der Benutzerkennung TSOS zugeordnet.

Der Rechte- und Funktionsumfang der Benutzerkennung mit dem Privileg ACS-ADMINISTRATION ist im Abschnitt „ACS: Alias-Katalogsystem“ auf Seite 290 beschrieben.

### CUSTOMER-PRIVILEGE-1..8

Durch die Vergabe der Privilegien **CUSTOMER-PRIVILEGE-1** oder CUSTOMER-PRIVILEGE-2 oder CUSTOMER-PRIVILEGE-3 u.s.w. bis **CUSTOMER-PRIVILEGE-8** ist es der Systembetreuung möglich, den Zugang zu Kommandos und Anweisungen für bestimmte Benutzerkennungen flexibel zu gestalten.

Standardmäßig sind diese Privilegien an keine Benutzerkennung vergeben.

### File-Transfer-Verwaltung

Die File-Transfer-Verwaltung (das ist die Benutzerkennung mit dem Privileg **FT-ADMINISTRATION**) darf das Auftrags- und Netzbeschreibungsbuch des Software-Produkts openFT verwalten (siehe Handbuch „openFT“ [23]).

### FTAC-Verwaltung

Die FTAC-Verwaltung (das ist die Benutzerkennung mit dem Privileg **FTAC-ADMINISTRATION**) darf die Schutzfunktionen des Software-Produkts openFT-AC verwalten (siehe Handbuch „openFT“ [23]).

### Systemglobale Guard-Administration (GUARD-ADMINISTRATION)

Die systemglobale Guard-Administration mit dem Privileg **GUARD-ADMINISTRATION** hat das Recht, auf allen lokalen Pubsets Aktionen zur Verwaltung von Guards beliebigen Typs durchzuführen und mit dem Programm GUARDS-SAVE benutzerkennungsübergreifend Guards zu sichern und zu restaurieren. Das heißt, eine Benutzerkennung mit diesem Privileg ist Miteigentümer aller Guards im System.
Das Privileg ist standardmäßig der Benutzerkennung TSOS zugeordnet.

### Hardware-Online-Wartung

Das Privileg **HARDWARE-MAINTENANCE** beinhaltet das Recht zur Durchführung der Hardware-Online-Wartung. Es umfasst insbesondere folgende Funktionen:

- Führen der Hardware-Fehlerstatistikdatei
- Ablauf von Hardware-Test- und Diagnose-Programmen (TDP) sowie Statistik- und Verfolgungs-Programmen unter Steuerung von BS2000/OSD simultan zu den Benutzerprogrammen

Das Privileg ist standardmäßig der Benutzerkennung SERVICE zugeordnet.

Wird das Privileg HARDWARE-MAINTENANCE an eine beliebige Benutzerkennung vergeben, ist Folgendes zu beachten:

- Eine Benutzerkennung mit dem Privileg HARDWARE-MAINTENANCE ist nur dann zum Zugriff auf Dateien fremder Benutzerkennungen berechtigt, wenn Folgendes gilt:
  - Falls die Datei mit Guards geschützt ist, müssen im Guard Zugriffsbedingungen festgelegt sein, die der privilegierten Benutzerkennung den Zugriff erlauben.
  - Falls die Datei nicht mit Guards, aber durch eine einfache Zugriffskontroll-Liste geschützt ist, muss diese der privilegierten Benutzerkennung den Zugriff erlauben.
  - Wenn die Datei weder mit Guards noch mit einer BACL geschützt ist, muss USER-ACCESS=*SPECIAL gesetzt sein.

  Es muss dafür gesorgt werden, dass diese Benutzerkennung alle benötigten Zugriffsrechte hat.
- Benutzerkennungen mit dem Privileg HARDWARE-MAINTENANCE unterliegen aus Sicherheitsgründen besonderen Einschränkungen. Insbesondere das Laden und Ausführen von Programmen ist nicht allgemein gewährleistet.
- Einige der bei der Hardware-Online-Wartung verwendeten Wartungsprogramme sind nicht erfolgreich ausführbar, sondern nur unter der Benutzerkennung SERVICE ablauffähig, sofern diese das Privileg HARDWARE-MAINTENANCE besitzt.

Weitere Informationen über die Hardware-Online-Auswertung finden Sie im „Diagnosehandbuch“ [14].

### HSMS-Verwaltung

Die Benutzerkennung mit dem Privileg **HSMS-ADMINISTRATION** darf in BS2000/OSD Aktionen zur Verwaltung des „Hierarchical Storage Management System“ durchführen (siehe Handbuch „HSMS“ [24]).

Das Privileg ist standardmäßig den Benutzerkennungen TSOS und SYSHSMS zugeordnet.

### Netzverwaltung

Ein Benutzer-Job mit dem Privileg **NET-ADMINISTRATION** darf Netzverwaltungsfunktionen ausüben, u.a. die Redefinition von Stationsnamen.

Das Privileg ist standardmäßig der Benutzerkennung TSOS zugeordnet.

### Notification-Service-Administration

Das Privileg **NOTIFICATION-ADMINISTRATION** berechtigt zur Konfiguration des Notification Service, d. h. es erlaubt die Produkte festzulegen, die den Notification Service nutzen dürfen und welche Methoden zur Benachrichtigung dabei unterstützt werden.

Das Privileg ist standardmäßig den Benutzerkennungen TSOS und SYSSNS zugeordnet.

Notification Service in BS2000/OSD ist ein Mechanismus, mit dem Benutzer beim Auftreten bestimmter Ereignisse benachrichtigt werden können. Derzeit wird diese Funktionalität von SPOOL genutzt. Ein Benutzer kann per Mail benachrichtigt werden, wenn bei seinen Druckaufträgen bestimmte Ereignisse, z.B. Job-Beendigung, eintreten. Siehe auch das Handbuch „SNS“ [53].

### Operating

Das Privileg **OPERATING** erlaubt, Aufgaben des Operators wahrzunehmen. Somit können Operator-Funktionen auch von Benutzerarbeitsplätzen ausgeführt werden.
Zur Basissystembedienung ist jedoch die Bedienung von (der normalen) Konsole weiterhin unablässig.

Das Privileg ist standardmäßig der Benutzerkennung SYSOPR zugeordnet.
Siehe auch Abschnitt „Privilegien für das Operating“ auf Seite 515.

### POSIX-Benutzerverwaltung

Das Privileg **POSIX-ADMINISTRATION** schützt den Zugriff auf POSIX-Attribute, die durch die BS2000-Benutzerverwaltung verwaltet werden, sowie das Tool zum Installieren zusätzlicher Produkte in POSIX. Die POSIX-Attribute können u.a. folgendermaßen geschützt werden:

- durch die Verwaltung der POSIX-Benutzerattribute aller Benutzerkennungen auf allen lokalen Pubsets. Diese Berechtigung stellt eine Untermenge des Privilegs USER-ADMINISTRATION dar.
- durch die Nutzung privilegierter Teilfunktionen des POSIX-SVC

Das Privileg ist standardmäßig der Benutzerkennung SYSROOT zugeordnet und kann dieser nicht entzogen werden.

Siehe auch Kapitel „POSIX-Benutzerverwaltung“ auf Seite 275.

### SPOOL-Verwaltung

Das Privileg **PRINT-SERVICE-ADMINISTRATION** erlaubt, SPOOL-Verwaltungsaufgaben durchzuführen. Dazu gehört:

- Start und Stopp von SPOOL-Geräten wie Drucker oder Bänder
- Veränderung der SPOOL-Parameter mit dem Dienstprogramm SPSERVE
- Veränderung von Drucksteuerdateien mit dem Dienstprogramm PRM
- Verteilter Zugriff auf vernetzte Drucker mit Dprint
- Verwaltung der Druckaufträge aller Benutzer

Das Privileg ist standardmäßig den Benutzerkennungen TSOS, SYSSNS und SYSSPOOL zugeordnet.

### PROP-XT-Verwaltung

Das Privileg **PROP-ADMINISTRATION** erlaubt, PROP-XT-Systemkommandos auszuführen. Zur PROP-XT-Verwaltung siehe das Handbuch „PROP-XT“ [42].

Das Privileg ist standardmäßig der Benutzerkennung TSOS zugeordnet.

### Auswertung der Audit-Dateien

Das Privileg **SAT-FILE-EVALUATION** gewährt das Recht, die SATLOG-Dateien und die CONSLOG-Dateien auszuwerten. Für den Inhaber des Privilegs SAT-FILE-EVALUATION ist die SAT-Protokollierung standardmäßig eingeschaltet, kann aber explizit wieder abgeschaltet werden.

Das Privileg ist standardmäßig der Benutzerkennung SYSAUDIT zugeordnet.

### Verwaltung der Audit-Dateien

Die Audit-Datei-Verwaltung benötigt das Privileg **SAT-FILE-MANAGEMENT** und darf folgende Aktionen durchführen:

- die von SAT (Security Audit Trail) erzeugten Dateien verwalten (insbesondere die SATLOG-Datei mit dem Kommando CHANGE-SAT-FILE umschalten)
- mit dem Kommando SET-REPLOG-READ-MARK den aktuellen Stand der Rep-Logging-Datei anfordern
- Informationen über den aktuellen Stand der EVENTING-Auswahl einholen
- die SATLOG-Dateien und die CONSLOG-Dateien auswerten

Für den Inhaber des Privilegs SAT-FILE-MANAGEMENT ist die SAT-Protokollierung standardmäßig eingeschaltet, kann aber explizit wieder abgeschaltet werden.
Das Privileg ist standardmäßig der Benutzerkennung SYSAUDIT zugeordnet.

### Sicherheitsverwaltung

Die Sicherheitsverwaltung (der Sicherheitsbeauftragte) hat das Recht zur Privilegienverwaltung, zur Verwaltung der Operator-Rollen und zum Schalten der Protokollierung (Auditing) mit SAT. Für die Benutzerkennung mit dem Privileg **SECURITY-ADMINISTRATION** ist die Protokollierung mit SAT immer eingeschaltet und kann nicht abgeschaltet werden.

Die Privilegienverwaltung darf die einzelnen Privilegien verwalten, d.h.

- Vergabe von Privilegien an Benutzerkennungen auf allen Pubsets
- Entzug von Privilegien für Benutzerkennungen auf allen Pubsets
- Vergabe und Entzug von Sammelprivilegien auf allen Pubsets
- Abfrage von Informationen über Sammelprivilegien

Hinsichtlich der Verwaltung der Operator-Rollen kann der Sicherheitsbeauftragte:

- Operator-Rollen definieren, modifizieren und löschen
- Operator-Rollen an Benutzerkennungen vergeben oder entziehen
- Informationen über die aktuelle Definition und Verteilung der Operator-Rollen abfragen

Das Privileg SECURITY-ADMINISTRATION kann nicht per Kommando an eine Benutzerkennung vergeben bzw. einer Benutzerkennung entzogen werden. Das Privileg ist in einem System ohne SECOS fest an die Benutzerkennung SYSPRIV geknüpft (die wie TSOS immer im Benutzerkatalog SYSSRPM enthalten ist), in einem System mit SECOS kann eine beliebige Benutzerkennung über den Startup-Parameterservice mit dem Privileg ausgestattet werden (siehe SECOS-Handbuch „Zugangs- und Zugriffskontrolle“ [48]).

### Ausführen von Benutzerkommandos

Mit dem Privileg **STD-PROCESSING** wird das Recht vergeben, die im Handbuch „Kommandos“ [27] beschriebenen Kommandos sowie die nichtprivilegierten Kommandos von Software-Produkten, die nicht zu BS2000/OSD-BC gehören, auszuführen.

Beim Neueinrichten einer Benutzerkennung (mit ADD-USER) wird dieser Benutzerkennung standardmäßig das Privileg STD-PROCESSING zugewiesen. Es ist außerdem standardmäßig an die vom System bei First-Start erzeugten Benutzerkennungen vergeben, mit Ausnahme der Benutzerkennungen SERVICE, SYSAUDIT und SYSPRIV.

Obwohl gilt, dass eine Benutzerkennung, die mit einem Privileg ausgestattet ist, nicht gelöscht werden kann, wird beim Löschen einer Benutzerkennung (mit REMOVE-USER) der „Besitz“ des Privilegs STD-PROCESSING nicht berücksichtigt, d.h eine Benutzerkennung kann gelöscht werden, wenn ihr außer diesem Privileg kein anderes mehr zugeordnet ist.

### Subsystem-Verwaltung

Die mit dem Privileg **SUBSYSTEM-MANAGEMENT** ausgestattete Benutzerkennung darf die Aufgaben zur globalen Verwaltung der Subsysteme der Dynamischen Subsystemverwaltung **DSSM** ausführen.

Das Privileg ist standardmäßig der Benutzerkennung TSOS zugeordnet.

### Software-Monitor-Verwaltung

Eine Benutzerkennung, die mit dem Privileg **SW-MONITOR-ADMINISTRATION** ausgestattet ist, darf die Software-Monitore openSM2 und COSMOS starten, beenden und administrieren (siehe Handbuch „openSM2“ [51]).

Das Privileg ist standardmäßig der Benutzerkennung TSOS zugeordnet.

### Bandverwaltung

Die Bandverwaltung (das ist die Benutzerkennung, die mit dem Privileg **TAPE-ADMINISTRATION** ausgestattet ist) darf die Administrationsfunktionen des Bandverwaltungssystems MAREN ausführen. D.h. sie darf das MAREN-Verwaltungsprogramm aufrufen, mit dem das MAREN-Archiv verwaltet werden kann (siehe Handbuch „MAREN“, Band 1 [31]).

Das Privileg ist standardmäßig den Benutzerkennungen TSOS und SYSMAREN zugeordnet.

### Encryption-Key-Verwaltung für Bänder

Die Encryption-Key-Verwaltung (das ist die Benutzerkennung, die mit dem Privileg **TAPE-KEY-ADMINISTRATION** ausgestattet ist) darf die Anweisungen des Programms MARENEKM (MAREN Encryption Key Manager) ausführen. D.h. sie darf die Encryption Keys für Bänder verwalten (siehe Handbuch „MAREN“, Band 1 [31]).

Das Privileg ist standardmäßig der Benutzerkennung SYSMAREN zugeordnet.

### TSOS

Das Privileg **TSOS** erlaubt es der Benutzerkennung TSOS, Systemverwaltungsfunktionen auszuführen.

Das Privileg TSOS ist fest an die Benutzerkennung TSOS geknüpft und kann weder dieser Benutzerkennung entzogen noch an eine andere Benutzerkennung vergeben werden.

### Benutzerverwaltung

Die systemglobale Benutzerverwaltung (das sind alle mit dem Privileg **USER-ADMINISTRATION** ausgestatteten Benutzerkennungen) darf auf allen lokalen Pubsets (für alle Benutzer oder Benutzergruppen) eine Aktion zur Benutzer- oder Benutzergruppenverwaltung durchführen.
Für die Vergabe von Betriebsmitteln und Rechten des Benutzerkatalog-Eintrags (wie z.B. START-IMMEDIATE, NO-CPU-LIMIT,...) an Benutzerkennungen und Benutzergruppen gibt es keine Begrenzungen.

Das Privileg ist standardmäßig der Benutzerkennung TSOS zugeordnet.

### VM-Verwaltung

Das Privileg **VIRTUAL-MACHINE-ADMINISTRATION** erlaubt, eine virtuelle Maschine (VM) mit Ausnahme der Monitor-VM zu bedienen. Es dürfen also VM-Administrator-Kommandos für die eigene VM gegeben werden (siehe Handbuch „VM2000“ [62]).

Das Privileg ist standardmäßig der Benutzerkennung TSOS zugeordnet.

### VM2000-Verwaltung

Das Privileg **VM2000-ADMINISTRATION** erlaubt, die Monitor-VM zu administrieren und sämtliche VM2000-Kommandos einzugeben (siehe Handbuch „VM2000“ [62]).

Das Privileg ist standardmäßig der Benutzerkennung TSOS zugeordnet.

## 12.4.4 Privilegienverteilung

Fährt man ein System mit First-Start hoch, wird der Benutzerkatalog SYSSRPM neu erzeugt. Standardmäßig haben vordefinierte Benutzerkennungen bestimmte Privilegien:

| Privileg \ Benutzerkennungen | TSOS | SERVICE | SYSAUDIT | SYSHSMS | SYSMAREN | SYSOPR | SYSPRIV [1] | SYSROOT | SYSSNS | SYSSPOOL | Andere [2] |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ACS-ADMINISTRATION | X [3] | - [4] | - | - | - | - | - | - | - | - | - |
| CUSTOMER-PRIVILEGE-1...8 | - | - | - | - | - | - | - | - | - | - | - |
| FT-ADMINISTRATION | X | - | - | - | - | - | - | - | - | - | - |
| FTAC-ADMINISTRATION | X | - | - | - | - | - | - | - | - | - | - |
| GUARD-ADMINISTRATION | X | - | - | - | - | - | - | - | - | - | - |
| HARDWARE-MAINTENANCE | - | X | - | - | - | - | - | - | - | - | - |
| HSMS-ADMINISTRATION | X | - | - | X | - | - | - | - | - | - | - |
| NET-ADMINISTRATION | X | - | - | - | - | - | - | - | - | - | - |
| NOTIFICATION-ADMINISTRATION | X | - | - | - | - | - | - | - | X | - | - |
| OPERATING | - | - | - | - | - | X | - | - | - | - | - |
| POSIX-ADMINISTRATION | - | - | - | - | - | - | - | X | - | - | - |
| PRINT-SERVICE-ADMINISTRATION | X | - | - | - | - | - | - | - | X | X | - |
| PROP-ADMINISTRATION | X | - | - | - | - | - | - | - | - | - | - |
| SAT-FILE-EVALUATION | - | - | X | - | - | - | - | - | - | - | - |
| SAT-FILE-MANAGEMENT | - | - | X | - | - | - | - | - | - | - | - |
| SECURITY-ADMINISTRATION | - | - | - | - | - | - | X | - | - | - | - |
| STD-PROCESSING | X | - | - | X | X | X | - | X | X | X | X |
| SUBSYSTEM-MANAGEMENT | X | - | - | - | - | - | - | - | - | - | - |
| SW-MONITOR-ADMINISTRATION | X | - | - | - | - | - | - | - | - | - | - |
| TAPE-ADMINISTRATION | X | - | - | - | X | - | - | - | - | - | - |
| TAPE-KEY-ADMINISTRATION | - | - | - | - | X | - | - | - | - | - | - |
| TSOS | X | - | - | - | - | - | - | - | - | - | - |
| USER-ADMINISTRATION | X | - | - | - | - | - | - | - | - | - | - |
| VIRTUAL-MACHINE-ADMINISTRATION | X | - | - | - | - | - | - | - | - | - | - |
| VM2000-ADMINISTRATION | X | - | - | - | - | - | - | - | - | - | - |

Tabelle 40: Privilegienverteilung nach First Start (Standard-Privilegienverteilung)

[1] Wenn in der Startup-Parameterdatei eine Benutzerkennung ungleich SYSPRIV als Benutzerkennung des Sicherheitsbeauftragten angegeben wurde, so gilt diese Spalte für eben diese Benutzerkennung. Die Benutzerkennung SYSPRIV ist in diesem Fall unter „Andere“ einzuordnen.

[2] die Systemkennungen SYSDUMP, SYSGEN, SYSNAC, SYSSOPT, SYSSNAP, SYSUSER

[3] X bedeutet: Der Benutzerkennung ist das Privileg standardmäßig zugeordnet

[4] - bedeutet: Der Benutzerkennung ist das Privileg standardmäßig nicht zugeordnet

### Privilegienverteilung nach Nicht-First-Start mit SECOS

Die Standard-Privilegienverteilung kann nur mit SECOS geändert werden.

Erfolgt nach einem Shutdown in BS2000/OSD ein Startup im System derselben Version mit Kaltstart, Warmstart, SELECTIVE-Start oder ZIP-Start, so ist die Privilegienverteilung dieselbe wie vor dem letzten Shutdown.

Sofern mit dem Startup jedoch gleichzeitig erstmalig ein Versionswechsel aus einer kleineren Version in die aktuelle Version von BS2000/OSD-BC erfolgt (Versionshochstieg), werden neue Privilegien an die Benutzerkennungen verteilt, an die sie auch bei First-Start vergeben worden wären.

### Privilegienverteilung nach Pubset-Import

Die Privilegienverteilung nach einem Pubset-Import mit ACTUAL-JOIN=*FIRST entspricht der nach einem First-Start, ein Pubset-Import anderer Art der nach einem Nicht-First-Start.

Erfolgt gleichzeitig eine Rekonstruktion des Benutzerkatalogs, werden für alle die Benutzer die Privilegien rekonstruiert, die aus dem alten Benutzerkatalog (gesichert in der Datei $TSOS.SYSSRPM.BACKUP) in den neuen übernommen werden. Diesen Benutzern sind also nach Abschluss der Rekonstruktion dieselben Privilegien zugeordnet wie zum Zeitpunkt der Sicherung. Welche Benutzerkennungen letztendlich rekonstruiert werden, ist abhängig vom Rekonstruktionstyp.

Gehört der alte Benutzerkatalog zu einem System mit SECOS-Funktionalität und soll er auf einem System ohne SECOS rekonstruiert werden, so werden in den neuen Benutzerkatalog für alle Benutzer die entsprechenden Standardwerte für die Privilegierung eingetragen. Das entspricht einer Privilegienverteilung nach First-Start.

## 12.5 Betriebsmittel für Benutzer begrenzen

Die Benutzerverwaltung kann für Benutzerkennungen Vorgaben machen, um die Nutzung folgender Betriebsmittel zu begrenzen bzw. deren Mißbrauch zu verhindern:

- Nutzung der Plattenspeicherkapazität auf den Pubsets
- Nutzung des Hauptspeichers im Server
- Nutzung der Leistung des Servers (CPU-Kapazität)

Die benutzerbezogene Zuteilung dieser Betriebsmittel erfolgt jeweils auf Kommandoebene mit ADD-/MODIFY-USER.

Die tatsächliche Verteilungs-Steuerung und -Kontrolle dieser Betriebsmittel im Rahmen der vorgegebenen Werte erfolgt im Betriebssystem (z.B. Verwaltung der Taskkategorien, Steuerung durch PCS, Verwaltung von Job-Streams und Jobklassen,...).



Eine analoge Steuerung für Benutzergruppen ist mit SECOS möglich.

Die Nutzung der systemglobalen Betriebsmittel kann für Benutzerkennungen durch die in der folgenden Tabelle dargestellten Kommandos begrenzt werden.

| **genutzte Betriebsmittel** | | **Kommando** | **Operanden** |
|---|---|---|---|
| Plattenspeicherkapazität auf den Pubsets | Public-Space | ADD-USER/ MODIFY-USER-ATTRIBUTES | PUBSET=...,<br>PUBLIC-SPACE-LIMIT=...,<br>PUBLIC-SPACE-EXCESS=...,<br>FILE-NUMBER-LIMIT=....,<br>JV-NUMBER-LIMIT=...,<br>TEMP-FILE-SPACE=...,<br>DMS-TUNING-RESOURCES=... |
| Hauptspeicher im Server | Adressraum | ADD-USER/ MODIFY-USER-ATTRIBUTES | ADDRESS-SPACE-LIMIT=... |
| | Hauptspeicher | ADD-USER/ MODIFY-USER-ATTRIBUTES | RESIDENT-PAGES=... |
| | Task-(De-) Aktivierung | ADD-USER/ MODIFY-USER-ATTRIBUTES | ACCOUNT-ATTRIBUTES= (MAX-ALLOWED-CATEGORY=..., PRIVILEGE=NO / PARAMETERS (INHIBIT-DEACTIVATION=...)) |
| Leistung des Servers (CPU-Kapazität) | CPU-Limit | ADD-USER/ MODIFY-USER-ATTRIBUTES | ACCOUNT-ATTRIBUTES= (CPU-LIMIT=..., PRIVILEGE=NO/PAR- (NO-CPU LIMIT=...)) |
| | zulässige Ablaufpriorität | ADD-USER/ MODIFY-USER-ATTRIBUTES | ACCOUNT-ATTRIBUTES= (MAXIMUM-RUN-PRIORITY=...) |
| | zulässige Taskkategorien | ADD-USER/ MODIFY-USER-ATTRIBUTES | ACCOUNT-ATTRIBUTES= (MAX-ALLOWED-CATEGORY=...) |
| | Scheduling-Recht | ADD-USER/ MODIFY-USER-ATTRIBUTES | ACCOUNT-ATTRIBUTES= (PRIVILEGE=NO/PARAMETERS- (START- IMMEDIATE=...)) |
| | Performance-Maßnahmen | ADD-USER/ MODIFY-USER-ATTRIBUTES | DMS-TUNING-RESOURCES=... |

## 12.6 Erfüllung von Sicherheitsanforderungen durch SECOS

Mit dem Software-Produkt SECOS wird BS2000/OSD-BC um die Funktionen erweitert, die einen sicheren Betrieb nach den geforderten Sicherheitskriterien gewährleisten.

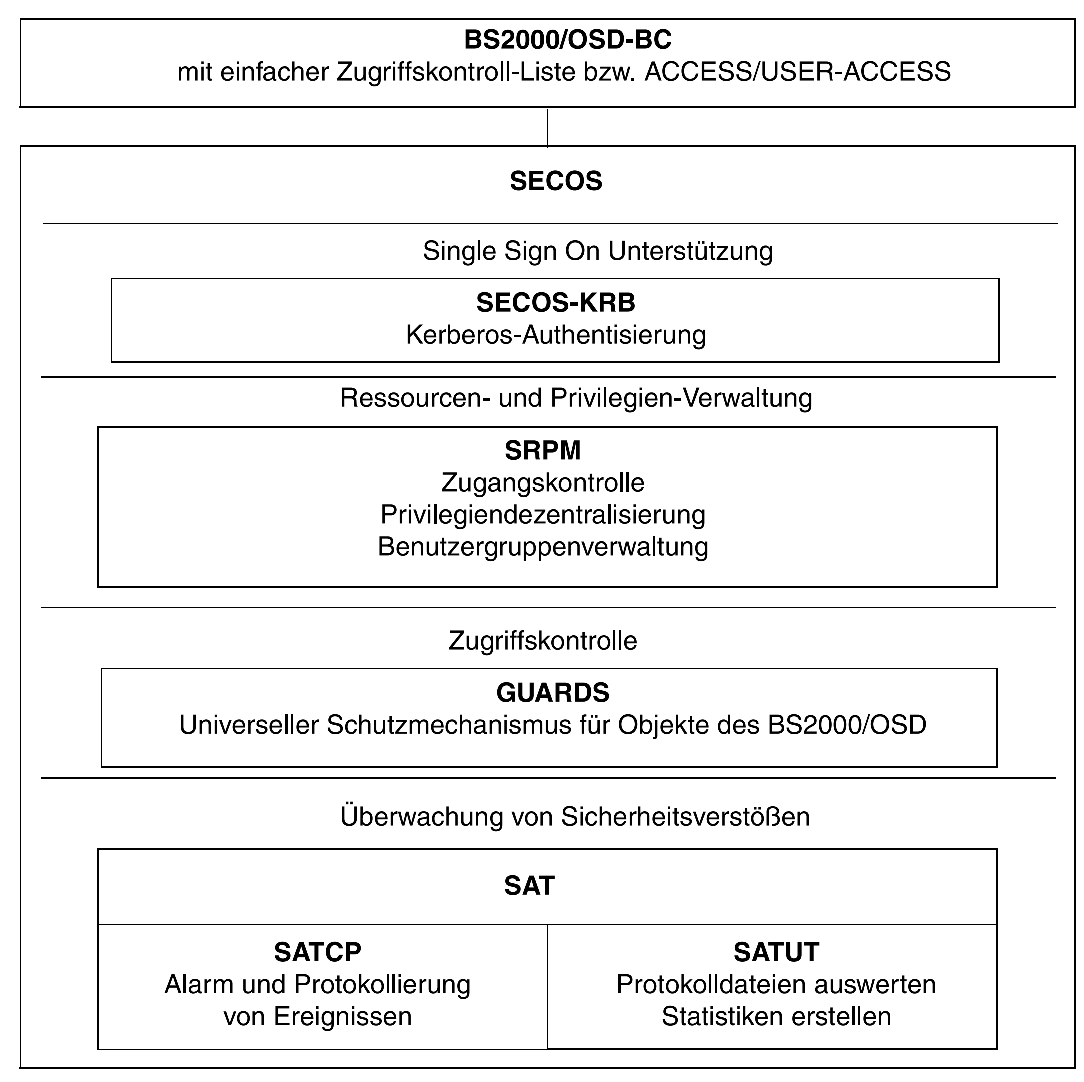



Bild 29: Funktionseinheiten des Sicherheitspakets

### Komponenten von SECOS

**SRPM** (System Resources and Privileges Management) bietet folgende Möglichkeiten:

- Dezentralisierung von Privilegien der Systembetreuung. Damit ist eine Bündelung oder Entbündelung von Aufgaben der Systembetreuung möglich, d.h. es kann eine dem jeweiligen Data Center angemessene Aufgabenverteilung vorgenommen werden. Die Privilegienvergabe erfolgt unter der Benutzerkennung des Sicherheitsbeauftragten (siehe Privileg SECURITY-ADMINISTRATION, Seite 521).
- Identifikation und Authentisierung von Benutzern durch erweiterten Zugangsschutz (Trennung und Kontrolle der Zugangswege, Terminal-Sets) und Kennwortschutz (Lebensdauerbegrenzung, minimale Länge, Komplexität).
- Einrichtung von Benutzergruppen, die einen differenzierten Zugriffsschutz ermöglichen. Zugleich können dadurch Verwaltungsaufgaben durch Gruppenverwalter dezentral wahrgenommen werden. Dies bedeutet eine Entlastung der Systembetreuung von trivialen Aufgaben.

**GUARDS** (General Usable Access Control Administration System) ermöglicht dem Benutzer, den Zugriff zu seinen Datenobjekten von sog. „Zugriffsbedingungen“ abhängig zu machen. Um seine Datenobjekte vor missbräuchlichem Zugriff zu schützen, legt der Benutzer Zugriffsbedingungen fest, die ein Subjekt erfüllen muss, bevor es auf die Datenobjekte zugreifen darf.

**GUARDDEF** (Default Protection, Standardschutz) dient der Vergabe von Standardattributwerten für Dateien und Jobvariablen. Diese Werte können wahlweise für das Anlegen oder Modifizieren dieser Objekte vorgegeben werden. Die Einstellungen können von der Systembetreuung (TSOS) jeweils pubset-weit und von jedem Benutzer für seine eigenen Objekte unter seiner Benutzerkennung vorgenommen werden.
GUARDDEF nutzt GUARDS zur Ablage der Einstellungen.

Mit **GUARDCOO** wird der Mechanismus zur Miteigentümerschaft (Co-owner Protection, Dateien und Jobvariablen können unter einer anderen Benutzerkennung angelegt und mitverwaltet werden) verfeinert.
Dabei kann die Miteigentümerschaft für unterschiedliche Namensbereiche der Objekte sowohl der Benutzerkennung TSOS entzogen als auch anderen Benutzerkennungen oder den Inhabern bestimmter Privilegien gewährt werden.
GUARDCOO nutzt GUARDS zur Ablage der Einstellungen.

**SAT** (Security Audit Trail) ermöglicht Beweissicherung durch Protokollierung sicherheitsrelevanter Ereignisse in eine besonders geschützte Datei (SATLOG-Datei). Ausschließlich der Sicherheitsbeauftragte hat das Recht, die SAT-Protokollierung zu aktivieren/deaktivieren und die SAT-Protokollierung für Benutzerkennungen und protokollierbare Ereignisse ein- bzw. auszuschalten.

Mit der Komponente SATUT lassen sich die SATLOG-Dateien von einem besonders autorisierten Benutzer mit den Privilegien SAT-FILE-MANAGEMENT oder SAT-FILE-EVALUATION auswerten. Die Auswertung kann so eingestellt werden, dass z.B. ein Rückblick auf spezielle Verarbeitungsschritte oder auf Aktionen von bestimmten Benutzerkennungen ausgegeben wird. Auf diese Weise lässt sich z.B. eine missbräuchliche Benutzung des Systems oder der unerlaubte Zugriff auf gesicherte Daten entdecken.

Für SSO (Single Sign On) im BS2000/OSD wird Kerberos eingesetzt.
Mit **SECOS-KRB** (Kerberos-Authentisierung) wird SSO im BS2000/OSD realisiert. Kerberos ist ein SSO-Sicherheitssystem (Single Sign On), das auf kryptographischen Verschlüsselungsverfahren basiert. Bei einer Authentisierung mit Kerberos werden keine Kennwörter im Klartext über das Netzwerk gesendet. Dadurch wird das Abfangen von Kennwörtern im Netzwerk verhindert.

Das Software-Produkt SECOS, seine einzelnen Komponenten und deren Installation und Integration in BS2000/OSD sind in den SECOS-Handbüchern [48] und [49] ausführlich beschrieben.

# 13 Datensicherung

Dieses Kapitel stellt Kriterien zur Auswahl von Sicherungskonzepten in BS2000/OSD, Datensicherungsfunktionen in BS2000/OSD-BC sowie Software-Produkte zur Datensicherung (ARCHIVE, HSMS und FDDRL) vor und informiert über Möglichkeiten zur Rekonstruktion von Dateien und Datenträgern.

Zum Schluss des Kapitels werden die von BS2000/OSD unterstützten Archivsysteme mit dem sie steuernden Software-Produkt ROBAR kurz beschrieben.

## 13.1 Kriterien zur Auswahl von Sicherungskonzepten

Jedes Data Center widmet dem Thema Datensicherung besondere Aufmerksamkeit. Die Notwendigkeit für ein lückenloses Sicherungskonzept ist begründet in

- der Forderung nach hoher Datenverfügbarkeit
- der Möglichkeit, auf Datenbestände zuzugreifen, die aus Kapazitätsgründen nicht auf gemeinschaftlichen Datenträgern gespeichert sind
- der Verbesserung der Antwortzeiten, indem sowohl gemeinschaftliche als auch private Datenträger reorganisiert werden
- dem Wunsch bzw. der Notwendigkeit, Datenbestände in ein anderes Data Center zu transportieren

Um diesen Forderungen gerecht zu werden, muss in jedem Data Center eine vorbeugende und regelmäßige Sicherung aller sicherungswürdigen Daten durchgeführt werden.

**Auswahl der Daten**

Die Auswahl der Daten, die bei einer Sicherung erfasst werden sollen, orientiert sich i.A. an ihrer Sicherungswürdigkeit. So kann unterschieden werden zwischen Produktionsdaten, die für die laufende Produktion erforderlich sind und einer steten Änderung unterliegen, und reinen Testdaten, die jederzeit reproduzierbar sind. Zudem können die Systemdaten, die sich im laufenden Betrieb normalerweise nicht ändern, von regelmäßigen Sicherungen ausgeklammert werden. Hier genügt eine vollständige Sicherungskopie, um im Rekonstruktionsfall den aktuellen Stand wieder herzustellen.

### Zeitpunkt und Häufigkeit der Sicherung

Das Sicherungsverfahren muss so konzipiert sein, dass es sowohl der Forderung nach Datensicherheit als auch der Forderung nach Verfügbarkeit der Anwendungen Rechnung trägt. So sollten z.B. langwährende Sicherungen – logische oder physikalische Gesamtsicherungen – gezielt in lastarme Zeiten verlegt werden, um den Ausfall an nutzbarer Rechenzeit zu minimieren.
Das Sicherungsverfahren ist ferner so zu wählen, dass redundante Sicherungsbestände vermieden werden. Das heißt, dass jeder Änderungsstand einer Datei nur einmal erfasst werden sollte.

### Sicherungsarten

- Gesamtsicherung

  Bei einer Gesamtsicherung werden alle durch sonstige Auswahlkriterien bestimmten Dateien erfasst, die zum Zeitpunkt der Sicherung geschlossen sind. Diese Dateien werden vollständig gesichert, unabhängig davon, ob sie sich seit der letzten Sicherung geändert haben oder nicht.
  Bei starker Zersplittung des Speicherplatzes bzw. in regelmäßigen Abständen sollte die Systembetreuung eine Reorganisation des Dateibestandes vornehmen. Dazu wird das Softwareprodukt SPACEOPT angeboten, siehe Abschnitt „Reorganisation von Plattenspeicherplatz“ auf Seite 397.

  Die Gesamtsicherung wird oft auch „Vollsicherung“ genannt.
  Jede physikalische Sicherung ist stets eine Gesamtsicherung.

- Differenzsicherung

  Bei der Differenzsicherung werden nur die Dateien gesichert, die sich seit der letzten Sicherung geändert haben oder die neu erstellt wurden. Diese Dateien werden vollständig gesichert. Voraussetzung ist, dass mit der gleichen Directory-Datei gearbeitet wird, in der die entsprechenden Informationen bezüglich der Dateiversion hinterlegt sind.

- partielle Sicherung

  Bei der partiellen Sicherung (einer besonderen Form der Differenzsicherung) wird für ausgewählte Dateien nur eine Sicherung der PAM-Seiten durchgeführt, die nach der letzten vollständigen Sicherung der Datei verändert wurden. Die anderen Dateien werden bei dieser Sicherung überhaupt nicht gesichert.
  Zur Rekonstruktion einer partiell gesicherten Datei sind jeweils die letzte partielle Sicherung und die letzte Gesamtsicherung erforderlich.

  Die partielle Sicherung wird oft auch „Teilsicherung“ genannt.

### Umfang der Sicherung

Den Umfang der Sicherung bestimmt die Systembetreuung abhängig von den Kriterien Datenbestand, Datenkonsistenz, Server-Last und Konfiguration.

- Datenbestand

  Die Anzahl und Größe der zu verwaltenden und zu sichernden Daten beeinflusst den Umfang der Sicherung.
  Bei einem geringen Datenbestand kann die Systembetreuung auf Teil- und Differenzsicherungen verzichten und regelmäßig eine logische oder physikalische Gesamtsicherung des Systems vornehmen. Dabei werden zwar die unveränderten Daten auch immer wieder gesichert, das gesamte „Datenpaket“ ist aber ständig konsistent und muss bei Bedarf nicht aus verschiedenen Sicherungsständen rekonstruiert werden.
  Bei umfangreichen Datenbeständen muss die Systembetreuung ein ausgefeilteres Konzept zur Datensicherung erarbeiten. Hierbei kann eine Sicherung aller Dateien sukzessive erfolgen, d.h. durch Aneinanderreihung mehrerer partieller und Differenzsicherungen, und in z.B. wöchentlichen Rhythmen eine logische oder physikalische Gesamtsicherung durchgeführt werden.
  Dateien können verschiedenen Backup-Levels oder Management-Klassen zugeordnet werden. Diese Zuordnungen können dann von der Systembetreuung zur Eingrenzung der Sicherung verwendet werden.

- Datenkonsistenz

  Der Datenbestand muss im Hinblick auf den Umfang der Sicherung auch einer qualitativen Überprüfung unterzogen werden.

  Qualitative Merkmale der System- und Benutzerdateien sind z.B.:

  – Anzahl der Zugriffe
  – Häufigkeit der Änderungen
  – Umfang der Änderungen
  – Austauschhäufigkeit von Programmversionen
  – Zuordnung zu einem Backup-Level oder einer Management-Klasse

  Beschränkt sich der Datenbestand im Wesentlichen auf konstante, stabile Versionen, die selten oder nicht verändert werden, kann der aktuelle Stand auch aus einer länger zurückliegenden Sicherungsversion schnell rekonstruiert werden. Bei häufig wechselnden Programmversionen und rasch wechselndem Dateienbestand muss die Systembetreuung einen Datenverlust durch ein entsprechend abgestuftes Sicherungskonzept verhindern.

- Server-Last

  Durch eine genaue Analyse der Server-Last kann die Systembetreuung nicht nur Zeitpunkt und Häufigkeit der Sicherungen bestimmen, sondern auch Rückschlüsse auf den Umfang der zu sichernden Daten ziehen.
  Das Sicherungskonzept könnte idealerweise so gewählt werden, dass in einer relativ lastfreien Periode eine umfangreiche Vollsicherung, in einer Periode mittlerer Last partielle und Differenzsicherungen durchgeführt werden, unter Voll-Last jedoch auf eine Sicherung gänzlich verzichtet wird.

- Konfiguration

  Die Konfiguration des Servers und damit auch die für die Datensicherung verfügbare Hardware beeinflusst die Erstellung des Sicherungskonzeptes und den Umfang der einzelnen Sicherungen.
  Stehen der Systembetreuung genügend periphere Geräte zur Verfügung, kann dadurch die Sicherungsdauer verkürzt und die Menge der Sicherungsdaten erhöht werden.

  – Durch Nutzung der Subtask-Funktion der Dienstprogramme ARCHIVE und FDDRL können Teilaufträge auf verschiedene Geräte gelenkt und die Sicherungen parallel abgewickelt werden.

  – Die Verteilung der Nutzerkreise auf einzelne Pubsets (MPVS) erleichtert den gezielten Zugriff auf Teilmengen der zu sichernden Daten.

  – Bereits bei der Generierung des Systems kann die Systembetreuung durch eine performance-günstige Konstellation „Peripherie zu Kanäle zu IOP“ die Voraussetzung für die beschleunigte Sicherung auch umfangreicher Datenmengen schaffen.

### Ausfallzeit durch Sicherung

Bei der Sicherung der Daten einer Anwendung ist die Anwendung selber für eine bestimmte Zeit nicht verfügbar. Diese Ausfallzeit wird vor allem durch Umfang und Zeitdauer der Sicherung bestimmt. Durch die Sicherung von abtrennbaren Kopien (z.B. auf Spiegelplatten) kann die Ausfallzeit auf das Erstellen der Kopie verkürzt werden. Die Anwendung selber kann nach der kurzen Kopierphase auf den Originaldateien weiterarbeiten, während die zeitaufwendigere Sicherung von der abgetrennten Kopie erfolgt.

### Logische und physikalische Sicherung

Bei einer **logischen Datensicherung** werden einzelne Dateien und Jobvariablen von einem oder mehreren Datenträgern gelesen und zusammenhängend, also in logischen Einheiten, auf andere Datenträger geschrieben.

Für die logische Datensicherung stehen in BS2000/OSD die Software-Produkte ARCHIVE (siehe Handbuch „ARCHIVE“ [3]) und HSMS (siehe Handbuch „HSMS“ [24]) zur Verfügung.

HSMS bietet dem Benutzer die vier Grundfunktionen:

- Datensicherung (Backup)
- Langzeitarchivierung (Archival)
- Verdrängung (Migration)
- Datentransfer (Export/Import)

HSMS setzt dabei auf dem Software-Produkt ARCHIVE auf. Die meisten Funktionen, die früher über ARCHIVE aufgerufen wurden, stehen in HSMS kompatibel zur Verfügung.

Zur Verkürzung der Ausfallzeit wird in HSMS die Backup-Sicherung mit CCOPY (Concurrent Copy) von einer Kopie geboten.

Bei einer **physikalischen Datensicherung** werden keine einzelnen Dateien gesichert, sondern ganze Datenträger gemäß ihrer physikalischen Struktur. Dabei werden sämtliche Dateien eines Datenträgers, einschließlich der Datenträgerkennsätze, blockweise in der physikalischen Reihenfolge auf einen zweiten Datenträger geschrieben. Dieser ist dann identisch mit dem Originaldatenträger.
Für die physikalische Datensicherung steht in BS2000/OSD das Software-Produkt FDDRL (siehe Handbuch „FDDRL“ [22]) zur Verfügung.

Eine Mischung von physikalischer und logischer Sicherung stellt die **Snapset-Sicherung** dar: Sie ist eine Pubset-Sicherung, bei der für jede Pubset-Platte eine entsprechende Kopie auf einer Snap-Unit erstellt wird. Aus dieser Pubset-Sicherung können einzelne Dateien und Jobvariablen als logische Einheiten gelesen werden. Abhängig vom Plattenspeichersystem kann auch der gesamte Pubset rekonstruiert werden. Die Funktionen für die Snapset-Sicherung und den Zugriff auf die gesicherten Daten (auf Ebene von Pubset, Datei und Jobvariable) stehen in BS2000/OSD-BC zur Verfügung.

Verschlüsselte Dateien werden bei allen Sicherungsarten in verschlüsselter Form abgespeichert.

Die grundsätzlichen Unterschiede bei der Sicherung von Datenträgern (physikalische Sicherung), der Sicherung von Dateien und Jobvariablen (logische Sicherung) und der Snapset-Sicherung zeigt folgende Übersicht:

| | **Logische Sicherung** | **Physikalische Sicherung** | **Snapset-Sicherung** |
|---|---|---|---|
| Was wird gesichert? | Dateien, Katalogeinträge, Jobvariablen | komplette Datenträger, d.h. private und gemeinschaftliche Platten | kompletter Pubset (plattenweise) |
| Wer sichert? | Systembetreuung bzw. HSMS-Adminstration: sämtliche Benutzer- und Systemdateien; Benutzer: nur seine eigenen Dateien | Systembetreuung bzw. HSMS-Adminstration | Systembetreuung bzw. HSMS-Adminstration |
| Wann wird gesichert? | regelmäßig bei gestoppter Anwendung | regelmäßig bei excatiertem Pubset | regelmäßig im laufenden Betrieb (importierter Pubset) |
| Hilfsmittel | Dienstprogramme ARCHIVE und HSMS | Dienstprogramm FDDRL | Snapset-Kommandos in BS2000/OSD-BC |
| Besonderheiten | – Voll- und Differenzsicherung; komfortable Methoden zur Objektauswahl<br>– Verkürzung der Ausfallzeit durch Sichern von Additional-Mirror- bzw. Clone-Units | – Sichern nur der belegten Blöcke<br>– Verkürzung der Ausfallzeit durch Sichern von Additional-Mirror- bzw. Clone-Units | – Zugriff auf gesicherte Dateien, Katalogeinträge, Jobvariablen (zur Information bzw. Restaurierung) ist „online“ möglich für alle Benutzer gemäß ihrer DVS-Zugriffsrechte<br>– für Systembetreuung Restaurierung des kompletten Pubsets<br>– Keine Ausfallzeit<br>– Sicherung/Restaurierung im Minutenzeitraum |
| Sicherungsdatenträger | MBK, Platte und Net-Storage-Volume | MBK und Platte | Snap-Units (im Plattenspeichersystem) |

Tabelle 41: Logische und physikalische Sicherung

## 13.2 Datensicherung mit Snapsets

BS2000/OSD-BC bietet Datensicherungsfunktionen für die Sicherung auf sogenannte Snapsets.

BS2000/OSD unterstützt auch die Datensicherung durch Replikationsfunktionen externer Plattenspeichersysteme. Siehe dazu das Handbuch „SHC-OSD“ [50].

### Prinzip der Snapsets

Ein Snapset ist die Sicherung eines Pubsets auf so genannte Snap-Units eines Plattenspeichersystems. Ein Snapset dient der Wiederherstellung verlorener Daten (z.B. nach versehentlichem Löschen). Auch der komplette Pubset kann auf den Stand zum Zeitpunkt der Erstellung des Snapsets zurückgesetzt werden. Es können sowohl SM- als auch SF-Pubsets gesichert werden. Snapsets werden im laufenden Pubset-Betrieb von der Systembetreuung oder vom HSMS-Administrator erzeugt und später bei Bedarf wieder gelöscht. Für einen Pubset können maximal 52 Snapsets existieren (sofern diese Maximalzahl vom Plattenspeichersystem unterstützt wird).

Die Pubset-Sicherung auf einen Snapset wird praktisch ohne Ein-/Ausgaben im Minutenzeitraum hergestellt. Innerhalb des Plattenspeichersystems besitzen Snap-Units keine Speicherzuordnung zu Laufwerken. Sie nutzen ein blockweises Differenzverfahren und benötigen nur für die vom Original abweichenden Blöcke eigenen Speicherplatz.

Dadurch wird die Sicherung auf Platte mit Snap-Technik auch wirtschaftlich vertretbar: Wenn z.B. von einem Pubset der Kapazität 100 GB an einem Tag 10 Sicherungen erstellt werden, dann würde mit einer Clone-Technik 1 TB zusätzlicher Plattenspeicherplatz benötigt werden. Bei der Nutzung von Snapsets hingegen hängt der benötigte zusätzliche Plattenspeicherplatz vom Änderungsvolumen des Pubsets ab. Bei einem Änderungsvolumen von 20% pro Tag entspricht dies in obigem Beispiel einem zusätzlichen Plattenspeicherplatz von nur 0,2 TB (20% * 10 Sicherungen * 100 GB).

Snapsets stehen ausschließlich für Restore-Zwecke zur Verfügung (d.h. nur Lesezugriff): Sie werden bei der Erzeugung automatisch in Betrieb genommen. Bereits vorhandene Snapsets werden in der Regel beim Importieren des zugehörigen Pubsets in Betrieb genommen.

Mit folgenden Funktionen können Benutzer auf Snapsets zugreifen:

- Informationen über vorhandene Snapsets (d.h. Pubset-Sicherungen) einholen
- Informationen über restaurierbare Dateien und Jobvariablen einholen
- Dateien und Jobvariablen restaurieren
- Pubset restaurieren (nur privilegierter Benutzer)

Folgende Software-Produkte ermöglichen ebenfalls den Zugriff zu Snapsets:

- LMS ermöglicht das Selektieren und Kopieren von Elementen aus PLAM-Bibliotheken auf Snapsets (siehe Handbuch „LMS“ [29]).
- Mit HSMS können Dateien und Jobvariablen von einem Snapset in ein Backup-Archiv übernommen werden (siehe „Sicherung vom Snapset in ein Backup-Archiv“ auf Seite 552 bzw. Handbuch „HSMS“ [24]).

Die Snapset-Funktionen sind im Shared-Pubset-Betrieb in allen beteiligten Systemen verfügbar. Die Voraussetzungen dazu (BS2000-Versionen der Pubset-Sharer und von SHC-OSD) sind in den aktuellen BS2000-Freigabemitteilungen beschrieben.

In BS2000/OSD können Sie bis zu 52 Snapsets verwenden. Außerdem werden zum Zugriff auf Snapsets Programmschnittstellen angeboten, die funktionell den zuvor erwähnten Kommando-Schnittstellen entsprechen (siehe Handbuch „DVS-Makros“ [20]).

VM2000 unterstützt die Nutzung von Snapsets auf Gastsystemen mit BS2000/OSD ab V7.0 (siehe Handbuch „VM2000“ [62]).

### Einsatzbereich von Snapsets

Snapsets dienen kurzfristig als Backup bei Verlust von Dateien bzw. Jobvariablen eines Pubsets oder bei Verlust des gesamten Pubsets. Für Langzeitsicherungen oder Datentransfer sind Snapsets nicht geeignet. Die bei Snapsets verwendeten Snap-Units sind lokale Kopien innerhalb eines Plattenspeichersystems und bieten daher auch keinen Schutz bei Ausfall der Plattenspeichersystem-Steuerung (Katastrophenschutz). Es wird daher dringend empfohlen, auch Snapsets mit HSMS zu sichern (siehe „Sicherung vom Snapset in ein Backup-Archiv“ auf Seite 552).

Privatplatten (bzw. Dateien auf Privatplatten) können nicht auf Snapsets gesichert werden.

Der Betrieb von Snapsets kann mit einer gleichzeitigen Nutzung von Plattenkopien auf Basis von lokalen Spiegeln im gleichen Pubset kombiniert werden. Die parallele Nutzung von weiteren Snaps ist abhängig vom eingesetzten Plattenspeichersystem. Ein Mischbetrieb von Snapsets (/CREATE-SNAPSET) und Snapshots (/START-SNAP-SESSION, /ACTIVATE-SNAP) ist z.B. bei ETERNUS DX nicht erlaubt.
Diese anderen Nutzungsarten sind in den Handbüchern „HSMS“ [24] und „SHC-OSD“ [50] beschrieben.

Einschränkungen:

- Die Erzeugung eines Snapsets ist nicht möglich, wenn sich ein Clone für diesen Pubset im Zustand RESTORED befindet.
- Die Restaurierung des Pubsets von einem Snapset ist nicht möglich, wenn für diesen Pubset noch eine Clone-Session aktiv ist.

### Schnittstellenübersicht zur Arbeit mit Snapsets

Alle Kommandos für den Snapset-Betrieb, die Sicherung auf Snapset bzw. die Restaurierung von Snapsets werden im SDF-Anwendungsbereich „SNAPSET“ angeboten. Diese Kommandos sind im Handbuch „Kommandos“ [27] beschrieben.

| **Kommando** | **Funktion** |
|---|---|
| ADAPT-SNAPSET-ACCESS [1] | passt den Zugriff auf Snapsets bei Remote-Spiegelung an |
| CHECK-SNAPSET-CONFIGURATION [2] | prüft die Snapset-Konfiguration und aktiviert Snapsets |
| CREATE-SNAPSET [2] | erzeugt einen Snapset |
| DELETE-SNAPSET [2] | löscht einen Snapset |
| LIST-FILE-FROM-SNAPSET [3] | informiert über Dateien auf einem Snapset |
| LIST-JV-FROM-SNAPSET [3] | informiert über Jobvariablen auf einem Snapset |
| RESTORE-FILE-FROM-SNAPSET [3] | restauriert Dateien auf Basis eines Snapsets |
| RESTORE-JV-FROM-SNAPSET [3] | restauriert Jobvariablen auf Basis eines Snapsets |
| RESTORE-PUBSET-FROM-SNAPSET [2] | restauriert den Pubset auf Basis eines Snapsets (nicht unterstützt für ETERNUS DX) |
| SET-PUBSET-ATTRIBUTES [2] | bestimmt die maximal erlaubte Anzahl von Snapsets für den Pubset (Operand SNAPSET-LIMIT) |
| SET-SNAPSET-PARAMETER [2] | vereinbart die Snapset-Umgebung |
| SHOW-PUBSET-ATTRIBUTES | informiert u.a. über das Snapset-Limit |
| SHOW-SNAPSET-CONFIGURATION | informiert über vorhandene Snapsets und die Snapset-Konfiguration |

Tabelle 42: Schnittstellenübersicht zur Arbeit mit Snapsets

[1] Privileg TSOS erforderlich

[2] Privileg TSOS oder HSMS-ADMINISTRATION erforderlich

[3] Für diese Kommandos gibt es auch eine Programmschnittstelle, siehe Handbuch „DVS-Makros“ [20].

### 13.2.1 Snapset-Betrieb vorbereiten

Den Snapset-Betrieb für einen Pubset aufzunehmen bedeutet, dass für den Pubset Kopien (Sicherungen) auf Snapsets erstellt werden, die für die Restauration von Dateien bzw. Jobvariablen oder des gesamten Pubsets zur Verfügung stehen.
Bevor der Snapset-Betrieb für einen Pubset aufgenommen werden kann, sind einmalige Vorbereitungen zu treffen.

Die Vorgehensweise soll an folgendem Beispiel für das Plattenspeichersystem Symmetrix veranschaulicht werden:

- Ausgangssituation

  Der Pubset `A` besteht aus 8 Platten (`PUBA00` bis `PUBA07`). Er soll an jedem Arbeitstag zweimal (z.B. mittags und abends) gesichert werden. Der Zugriff auf die Sicherungen der letzten 6 Arbeitstage soll gewährleistet sein.

- Bedarfsplanung

  Für 2 x 6 Pubset-Sicherungen müssen maximal 12 Snapsets zur Verfügung stehen. Dazu werden 8 x 12 = 96 Snap-Units benötigt.

- Betrieb von TimeFinder/Snap im Plattenspeichersystem Symmetrix vorbereiten

  Die Funktionen benötigen speziell konfigurierte Gerätetypen im Plattenspeichersystem, sogenannte Virtual-Devices und Save-Devices. Eine Gruppe von Save-Devices ist ein Save-Pool. Im Normalfall nutzen die Snap-Units der Snapsets den Default-Pool. Dieser Pool sollte vom Service so groß eingerichtet werden, dass er etwa die Hälfte vom Plattenplatz des Original-Pubsets speichern kann.

  Die Plattenspeichersysteme Symmetrix unterstützen auch das Einrichten verschiedener Save-Pools. Wenn z.B. für jeden Pubset mit Snapset-Betrieb ein eigener Save-Pool benutzt wird, ist die Snap-Speichernutzung der einzelnen Pubsets innerhalb des Plattenspeichersystems gegeneinander abgeschottet. Ein Ressourcen-Engpass innerhalb eines Save-Pools wirkt sich dann nur auf den jeweiligen Pubset aus.

- Snap-Units im Plattenspeichersystem Symmetrix bereitstellen

  Die Snap-Units müssen mit gleichem Typ und gleicher Größe wie die Original-Platten des Pubsets `A` im gleichen Plattenspeichersystem vom Service eingerichtet werden. Das bedeutet 12 Snap-Units wie die Platte `PUBA00`, 12 Snap-Units wie `PUBA01`, usw., und schließlich 12 Snap-Units wie `PUBA07`.
  Eine einheitliche Plattenausstattung im Pubset erleichtert die Planung.
  Wenn bereits eine Pubset-Erweiterung abzusehen ist, sollten auch dafür schon zusätzliche Snap-Units in geplanter Zahl eingerichtet werden.

- Snap-Units in BS2000/OSD konfigurieren

  Alle 96 Snap-Units müssen wie normale Platten des gleichen Typs in die I/O-Generierung der beteiligten BS2000-Systeme aufgenommen werden, praktisch also in alle Hardware-Generierungen, in denen auch die Original-Platten des Pubsets `A` eingetragen sind (siehe Handbuch „Systeminstallation“ [57]).
  An S-Servern können die Platten auch im laufenden Betrieb in die aktive I/O-Konfiguration aufgenommen werden.
  Die Aufnahme von emulierten Platten D3435 in die X2000-Konfiguration eines SQ-Servers ist im Handbuch „Bedienen und Verwalten“ [59] beschrieben.

  *Hinweis für VM2000:*
  VM2000 erkennt Snap-Units. Snap-Units werden automatisch beim Pubset-Import zugeschaltet und unter VM2000 implizit der VM zugeordnet, wenn das VM-Privileg AUTO-SNAP-ASSIGNMENT vorhanden ist. Eine VM erhält beim Einrichten mit dem Kommando CREATE-VM standardmäßig dieses Privileg. Es erlaubt dem Gastsystem auf der VM, sich die Snap-Units (bzw. die entsprechenden virtuellen Geräte) eines Snapsets implizit zuzuordnen, ohne dass VM und Gerät für die implizite Gerätezuordnung vorbereitet sind (d.h. das VM-Privileg ASSIGN-BY-GUEST ist diesem Fall nicht notwendig).

- Snapset-Limit setzen und Save-Pool zuweisen (in BS2000/OSD)

  Das Snapset-Limit ist im SVL der Pubres des SF-Pubsets bzw. im SVL der Volres des Control-Volume-Sets des SM-Pubsets eingetragen. Es legt fest, wieviele Snapsets maximal für den Pubset zulässig sind. Bei einem Pubset ohne Snapset-Betrieb liegt das Snapset-Limit bei 0.
  Da für den Pubset `A` maximal 12 Pubset-Kopien zur Verfügung stehen sollen, setzt die Systembetreuung das Snapset-Limit auf 12:

  ```
  /SET-PUBSET-ATTRIBUTES PUBSET=A,SNAPSET-LIMIT=12
  ```

  Sind die benötigten Save-Pools, also z.B. ein Save-Pool A für den Pubset A, durch den Service in der dafür notwendigen Größe eingerichtet worden, kann die Systembetreuung den speziellen Save-Pool zuweisen:

  ```
  /SET-SNAPSET-PARAMETER PUBSET=A,SAVE-POOL-NAME=A
  ```

### Snapset-Betrieb bei Katastrophenschutz mit entfernter Spiegelung

Wird der Pubset A wegen Katastrophenschutz in einem lokalen Plattenspeichersystem und in einem Remote-Plattenspeichersystem betrieben, sind folgende Fälle zu unterscheiden:

- Im Normalfall werden die Snapsets nur im lokalen Plattenspeichersystem eingerichtet. Sie sind somit lokale Pubset-Kopien, die bei Ausfall des lokalen Plattenspeichersystems im Remote-Plattenspeichersystem nicht zur Verfügung stehen.
  Sollen die Snapsets nicht in den Katastrophenschutz einbezogen werden, sind keine weiteren Vorbereitungen notwendig.

- Sollen die Snapsets im Katastrophenfall auch im Remote-Plattenspeichersystem zur Verfügung stehen, müssen sie in beiden Plattenspeichersystemen erzeugt werden. Dazu müssen im Remote-Plattenspeichersystem die gleiche Anzahl Snap-Units mit dem gleichen Save-Pool eingerichtet werden und die Snap-Units des Remote-Plattenspeichersystems in die gleichen Hardware-Generierungen aufgenommen werden. Zusätzlich muss die Systembetreuung in den Snapset-Parametern des Pubsets einstellen, dass Snapsets jeweils im lokalen und im Remote-Plattenspeichersystem erzeugt werden sollen:
  ```
  /SET-SNAPSET-PARAMETER PUBSET=A,REMOTE-COPY=*YES(RA-GROUP=*UNIQUE)
  ```

  In Symmetrix-Plattenspeichersystemen muss bei Spiegelung mit Concurrent-SRDF auf zwei Remote-Plattenspeichersysteme über die RA-Gruppe das Remote-Plattenspeichersystem bestimmt werden, in dem die Snapsets zusätzlich angelegt werden sollen:
  ```
  /SET-SNAPSET-PARAMETER PUBSET=A,REMOTE-COPY=*YES(RA-GROUP=<ra-group>)
  ```

  Das Kommando ADAPT-SNAPSET-ACCESS stellt sicher, dass die zugeordneten Snapsets nach einer solchen Umschaltung weiter zur Verfügung stehen, ohne dass der Pubset exportiert werden muss:
  ```
  /ADAPT-SNAPSET-ACCESS PUBSET=<cat-id>
  ```

## 13.2.2 Sicherungsbetrieb mit Snapsets

Sind die Vorbereitungen abgeschlossen, kann der Snapset-Betrieb (im Beispiel für den Pubset `A`) aufgenommen werden, d.h. die Systembetreuung bzw. die HSMS-Administration kann zu den geplanten Zeitpunkten Pubset-Kopien (=Snapsets) erstellen.

*Beispiel*

Pubset `A` soll an jedem Arbeitstag zweimal (mittags und abends) gesichert werden.

### Snapset erstellen

Jeweils zu dem gewünschten Sicherungszeitpunkt erstellt die Systembetreuung bzw. die HSMS-Administration mit dem Kommando CREATE-SNAPSET einen Snapset:

```
/CREATE-SNAPSET PUBSET=A
```

Das Kommando kann im Shared-Pubset-Betrieb von allen Systemen aus eingegeben werden. Die Voraussetzungen dazu (BS2000-Versionen der Pubset-Sharer und von SHC-OSD) sind in den aktuellen BS2000-Freigabemitteilungen beschrieben.
Zu einem Zeitpunkt kann für den gleichen Pubset nur ein CREATE-SNAPSET-Kommando gegeben werden. Parallele Aufrufe werden serialisiert.
Als Voreinstellung des Kommandos wird bei erreichtem Snapset-Limit implizit der älteste Snapset gelöscht. Optional kann im Operanden DELETE-EARLIEST auch vereinbart werden, dass in jedem Fall der älteste Snapset implizit gelöscht wird oder dass in keinem Fall implizit gelöscht wird.
Der neu erzeugte Snapset wird automatisch in Betrieb genommen und steht damit allen Benutzern für Restore-Zwecke zur Verfügung.

*Snapset-Identifikation*

Jeder neu erzeugte Snapset erhält eine eindeutige Snapset-Identifikation. Für die maximal 52 möglichen Snapsets werden fortlaufend in alphabetischer Reihenfolge zuerst die Kleinbuchstaben „a“ bis „z“, dann die Großbuchstaben „A“ bis „Z“ vergeben. Nach Erreichen von „Z“ beginnt die Vergabe wieder mit „a“ (die zeitliche Reihenfolge der Snapset-Erzeugung entspricht deshalb nicht immer der alphabetischen Reihenfolge der Snapset-Ids). An der Kommandoschnittstelle müssen die Snapset-Identifikationen entsprechend in Groß-/Kleinschreibung angegeben werden.
Jede Snapset-Unit erhält eine VSN, bei der in die VSN der Originalplatte die Snapset-Identifikation als Kleinbuchstabe eingesetzt wird:

– Bei einer VSN in Punkt-Notation wird der Punkt ersetzt und evtl. verschoben.
– Bei einer VSN in PUB-Notation werden das „U“ oder das „B“ der Zeichenfolge „PUB“ jeweils ersetzt.

Beispiel: Der dritte Snapset des Pubsets A erhält die Snap-Identifikation „c“ und besteht aus den Volumes `PccA00`, `PccA01`, ... `PccA07`.

### Verwaltung und Inbetriebnahme von Snapsets

Jeder neu erzeugte Snapset besitzt eine eindeutige Snapset-Identifikation (siehe Seite 543). Die Beschreibungen aller Snapsets eines Pubset werden im Snapset-Katalog hinterlegt und verwaltet. Der Snapset-Katalog wird im zugehörigen Pubset unter dem Namen $TSOS.SYSCAT.SNAPSET abgelegt und enthält folgende informationen:

– die Betriebsparameter für die Snapsets des Pubsets wie die Nutzung von Remote-Spiegelung und Angaben zum Save-Pool
– Beschreibung der einzelnen Snapsets mit Angaben wie Snapset-Identifikation, Erstellungsdatum und MNs der Snap-Units im lokalen Plattenspeichersystem und ggf. auch der Snap-Units im Remote-Plattenspeichersystem.

Der Snapset-Katalog wird, falls er noch nicht existiert, mit Erstellen des Snapsets eingerichtet. Der Snapset-Katalog liegt bei einem SM-Pubset auf der Volres des Control-Volume-Sets und bei einem SF-Pubset auf der zugehörigen Pubres.

*Snapsets in und außer Betrieb nehmen*

Bestehende Snapsets werden in der Regel beim Importieren des zugehörigen Pubsets in Betrieb genommen.

Vorhandene Snapsets werden nicht in Betrieb genommen, wenn das Subsystem SHC-OSD zum Zeitpunkt des Pubset-Imports noch nicht aktiv ist (insbesondere werden vorhandene Snapsets des Home-Pubsets beim Hochfahren des Systems nicht automatisch in Betrieb genommen). Sobald SHC-OSD aktiv ist, werden die Snapsets bei Aufruf des Kommandos SHOW-SNAPSET-CONFIGURATION nachträglich aktiviert.

Manche Snapsets können beim Import zwar in Betrieb genommen und aktiviert werden, verbleiben aber wegen Hardware- oder Konfigurationsproblemen im Zustand „nicht zugreifbar“. Diese Snapsets können nach Fehlerbehebung mit dem Kommando CHECK-SNAPSET-CONFIGURATION wieder zugreifbar gemacht werden.

Beim Neuanlegen eines Snapset erfolgt dies mit der CREATE-SNAPSET-Verarbeitung automatisch. Die Inbetriebnahme beinhaltet folgende Aktionen:

- Snapset-Katalog öffnen (durch GCF)
- alle Snap-Units zuweisen (ATTACH)
  In einer Konfiguration mit Remote-Spiegelung werden die „passenden“ Snap-Units zugewiesen. Im Normalzustand bei aktivem Remote-Spiegel sind das die Volumes in dem Plattenspeichersystem mit den Sources. Bei getrennten Spiegeln erfolgt die Auswahl anhand des importierten Original-Pubsets:
  – Beim Pubset am Source-Plattenspeichersystem werden nur die Volumes der Snapsets aus dem Source-Plattenspeichersystem attached und aktiviert.

– Beim Pubset am Target-Plattenspeichersystem werden nur Volumes der Snapsets aus dem Target-Plattenspeichersystem verwendet.

Unter VM2000 muss für ein Gastsystem mit Snapset-Betrieb der Parameter AUTO-SNAP-ASSIGNMENT=*YES gesetzt sein, damit die Snap-Units implizit der VM des Gastsystems zugeordnet werden (siehe Seite 541).

- CCOPY-Session für jeden Snapset aufbauen

Beim Exportieren des Pubsets werden die zugehörigen Snapsets außer Betrieb genommen. Die beim Importieren aufgeführten Aktionen werden in umgekehrter Reihenfolge durchgeführt.

### Snapset löschen

Unabhängig von der Snap-Erstellung kann die Systembetreuung bzw. die HSMS-Administration mit dem Kommando DELETE-SNAPSET auch explizit einen Snapset löschen:

```
/DELETE-SNAPSET PUBSET=A[,SNAPSET=*EARLIEST]
```

Hier wird der älteste Snapset des Pubsets `A` gelöscht.
Die Angabe `SNAPSET=*EARLIEST` kann auch entfallen (Voreinstellung).

Das Kommando kann im Shared-Pubset-Betrieb von allen Systemen aus eingegeben werden, die die entsprechende Snapshot-Funktionalität der angeschlossenen Plattenspeichersysteme unterstützen. Voraussetzung ist, dass der betroffene Pubset im lokalen System importiert ist und dass sowohl am Pubset-Master als auch im lokalen System BS2000/OSD ab V7.0 ablaufen.

Der zu löschende Snapset kann auch über seine Snapset-Identifikation oder als relative Angabe zum aktuellen Stand des Pubsets (mit -n, wobei -1 den jüngsten Snapset bezeichnet) angegeben werden. Mit dem Löschen eines Snapsets steht der entsprechende Sicherungsstand des Pubsets nicht mehr zur Verfügung. Gerade laufende Restore-Vorgänge für einen zu löschenden Snapset werden mit dem Löschen des Snapsets abgebrochen.

*Snapset-Betrieb beenden*

Mit der Angabe SNAPSET=*ALL werden alle Snapsets gelöscht und der Snapset-Betrieb für den Pubset beendet. Dieser Aufruf beinhaltet folgende Maßnahmen:

- alle Snapsets löschen
- Snapset-Katalog schließen und löschen
- Snapset-Limit des Pubsets auf Null setzen

### Snapsets anzeigen

Informationen über betriebsbereite Snapsets liefert das Kommando SHOW-SNAPSET-CONFIGURATION:

```
/SHOW-SNAPSET-CONFIGURATION PUBSET=A[,SNAPSET=*ALL]
```

Für jeden Snapset werden folgende Informationen ausgegeben:

- Erstellungsdatum
- Snapset-Identifikation
- Kennung der CCOPY-Session, über die der Zugriff auf den Snapset erfolgen kann

Die Systembetreuung bzw. die HSMS-Administration kann Informationen auch explizit für einen bestimmten Snapset anfordern. In diesem Fall werden folgende Informationen zusätzlich ausgegeben:

- Name eines Save-Pools, der dem Pubset zugeordnet ist
- Kennung des Remote-Plattenspeichersystems, wenn dort bei Einsatz von Remote-Spiegelung die Snapsets zusätzlich erstellt werden
- VSNs der Pubset-Volumes
- MNs der zugeordneten Snap-Units des lokalen Plattenspeichersystems (und ggf. auch die MNs des Remote-Plattenspeichersystems).

### Hinweise und Einschränkungen

*Kein Snapset-Betrieb*

Der Snapset-Betrieb für einen Pubset ist – wie alle anderen über SHC-OSD angebotenen Kopierfunktionen – nicht verträglich mit DAB-Schreib-Caching für diesen Pubset. Das Kommando CREATE-SNAPSET wird bei DAB-Schreib-Caching abgewiesen.

Snapset-Betrieb und DRV-Spiegelung für einen Pubset schließen sich gegenseitig aus: Auf einem Pubset mit Snapset-Betrieb (Snapset-Limit ungleich 0) kann kein DRV-Betrieb aufgenommen werden und umgekehrt.

*Snapset-Betrieb im Verbund mit BS2000/OSD bis V7.0*

Seit BS2000/OSD V8.0 gibt es zur Unterstützung von mehr als 26 Snapsets das „neue“ Format des Snapset-Katalogs mit Snapset-Identifikatoren in Klein- und Großbuchstaben. In BS2000/OSD V7.0 wurde das „alte“ Format des Snapset-Katalogs mit Snapset-Identifikatoren in Kleinbuchstaben verwendet.

In BS2000/OSD wird weiterhin das „alte“ Format des Snapset-Katalogs verwendet, wenn das Snapset-Limit ≤ 26 ist oder der Pubset im MSCF-Verbund gleichzeitig auch in Systemen mit BS2000/OSD V7.0 importiert ist.

Bei einem Snapset-Limit > 26 und wenn der Pubset im MSCF-Verbund gleichzeitig nur in Systemen mit BS2000/OSD ab V8.0 importiert ist, dann wird das alte Format bei /CREATE-SNAPSET automatisch in das neue Format konvertiert.

Eine Rückkonvertierung in das alte Format kann unter BS2000/OSD ab V8.0 erfolgen, wenn das Snapset-Limit ≤ 26 ist und Snapset-Identifikatoren lediglich in Kleinbuchstaben vorliegen.
Die Rückkonvertierung wird ausgelöst durch /DELETE-SNAPSET, /SET-SNAPSET-PARAMETER, /EXPORT-PUBSET oder /IMPORT-PUBSET (im Fehlerfall nach /SET-PUBSET-ATTRIBUTES, z.B. Systemausfall vor /DELETE-PUBSET oder /EXPORT-PUBSET).

*Snapset-Betrieb mit entfernter Spiegelung*

Wenn Snapsets auch auf dem Remote-Plattenspeichersystem betrieben werden sollen (siehe „Snapset-Betrieb bei Katastrophenschutz mit entfernter Spiegelung“ auf Seite 542), sind folgende Einschränkungen zu beachten:

- Die Snapset-Erzeugung auf dem Remote-Plattenspeichersystem ist nicht möglich, wenn die entfernte Spiegelung unterbrochen ist (entweder mit dem Kommando HOLD-REMOTE-COPY oder verursacht durch unterbrochene Links), wenn die entfernte Spiegelplatte noch nicht synchronisiert ist oder bei asynchroner entfernter Spiegelung.

- Wenn die Snapset-Erzeugung auf einem der beteiligten Plattenspeichersysteme (lokal oder remote) scheitert, so wird auch auf dem anderen Plattenspeichersystem kein Snapset erstellt.

- Wenn der Zugriff auf die Platten des Pubsets dynamisch zwischen Source- und Target-Controller umgeschaltet wird (durch die AutoSwap-Funktion von HIPLEX AF oder explizit durch SHC-OSD-Kommandos), dann wird der Zugriff auf die dem Pubset zugeordneten Snapsets nicht automatisch mit umgeschaltet.
  Das Kommando ADAPT-SNAPSET-ACCESS stellt sicher, dass die zugeordneten Snapsets nach einer solchen Umschaltung weiter zur Verfügung stehen, ohne dass der Pubset exportiert werden muss.

  Beim Kommandoaufruf wird geprüft, ob der Zugriff auf die zugeordneten Snapsets im gleichen Controller erfolgt wie der Zugriff auf die Platten des Pubsets. Wenn dies nicht der Fall ist, dann wird die Umschaltung für die dem Pubset zugeordneten Snapsets nachgebildet:
  - Die aktuell zugeschalteten Snapsets werden außer Betrieb genommen.
  - Anschließend werden die Snapsets des lokalen Controllers zugeschaltet.

*Betriebsmittelengpass und seine Folgen*

Wenn ein Save-Pool erschöpft ist, können in diesem Pool keine weiteren Snap-Units und damit auch kein weiterer Snapset erzeugt werden. Bereits bestehende Snap-Units wechseln aus SHC-OSD-Sicht in den Status FAILED und auf die dort abgelegten Daten kann nicht mehr zugegriffen werden. Damit sind die betroffenen Snapsets auch nicht mehr zugreifbar und ihre Sicherungsstände nicht mehr rekonstruierbar.

Damit auf einen bevorstehenden Überlauf des Save-Pools rechtzeitig reagiert werden kann, meldet SHC-OSD mit der Konsolmeldung NDE0055 das Überschreiten des voreingestellten Schwellwertes (Standard bei 80%, zu Abfrage und Einstellung siehe „SHC-OSD“ [50]). Gleichzeitig versucht DMS den ältesten Snapset des Save-Pools mit dem Kommando DELETE-SNAPSET zu löschen. Dazu muss der Pubset importiert sein.

*Änderung der Pubset-Konfiguration*

Bevor eine der folgenden Änderungen der Pubset-Konfiguration durchgeführt wird, muss der Snapset-Betrieb beendet werden (bedeutet Löschen aller Snapsets):

- vor dem Umbenennen eines Pubsets
- vor dem Überführen eines SF-Pubsets in einen Volume-Set eines SM-Pubsets
- vor dem Wechsel auf andere Pubset-Platten, d.h. beim Übergang von K- auf NK-Format, beim Wechsel von CKD- auf FBA-Platten oder bei einem physikalischen Umzug mit FDDRL (Save/Restore)

Bei Vergrösserung des Pubsets um weitere Platten muss auch die Menge der passenden Snap-Units entsprechend erweitert werden (siehe Abschnitt „Snapset-Betrieb vorbereiten“ auf Seite 540).

Bei der Reduzierung des Pubsets um Platten werden vorhandene Snapsets ungültig. Sie können nicht mehr zum Restaurieren herangezogen werden. Sie werden nach einer Neuinitialisierung der entfernten Platten beim nächsten Importieren des Pubset aus dem Snapset-Katalog entfernt. Vorhandene Snapsets müssen daher vor dem Herauskonfigurieren von Platten mit HSMS gesichert werden, falls die dort abgelegten Daten noch benötigt werden. Unmittelbar nach der Konfigurationsänderung sollte ein neuer Snapset erzeugt werden, damit die Möglichkeit zum Restaurieren von Dateien und Jobvariablen sowie des gesamten Pubsets erhalten bleibt.

*Vermeidung von hohem Speicherbedarf, Verhalten bei Speicherengpässen*

Heftiges und „breites“ Schreiben wirkt sich auf den Speicherbedarf des neuesten Snapsets aus und führt zu erhöhter Speichernutzung im zugeordneten Save-Pool. Folgende einfache Maßnahmen vermeiden besonders hohen Speicherbedarf im Save-Pool:

- keine Paging-Dateien in Pubsets mit Snapset-Betrieb erzeugen
- den Pubset möglichst wenig mit dem Softwareprodukt SPACEOPT reorganisieren

Wenn es im laufenden Betrieb zu einem Speicherengpass im Save-Pool kommt, löscht das System sofort den ältesten Snapset in diesem Pool und gibt zusätzliche Konsolmeldungen aus. Wenn die Ursache für diesen Engpass als Normalfall einzustufen ist, sollte zur Vermeidung derselben Situation in Zukunft der Save-Pool entsprechend vergrößert werden.

*Migrierte Dateien*

Damit migrierte Dateien nach der Restauration von Snapsets noch benutzbar sind, also der Recall noch funktioniert, müssen die zugehörigen Bänder noch vorhanden sein. Sie dürfen nicht freigegeben oder schon neu verwendet worden sein.
Wenn vorhandene Snapsets einen restaurierbaren Nahbereich von **n** zurückliegenden Tagen abdecken, sollte die Reorganisation der migrierten Dateien mit HSMS stets nur für einen Zeitraum vor diesen **n** Tagen durchgeführt werden.

### 13.2.3 Restaurierung von Dateien und Jobvariablen

Sobald der Snapset-Betrieb für einen Pubset aufgenommen wurde, stehen die Pubset-Kopien auf den erstellten Snapsets allen Benutzern für die Restaurierung zur Verfügung. Die Snapsets sind betriebsbereit, wenn der Pubset importiert ist. Der nichtprivilegierte Benutzer hat dabei Zugriff auf Dateien und Jobvariablen gemäß der DVS-Zugriffsrechte.

Informationen über alle betriebsbereiten Snapsets erhalten die Benutzer über das Kommando SHOW-SNAPSET-CONFIGURATION, wobei der privilegierte Benutzer auch genauere Informationen über Snapset-Platten erhält:

```
/SHOW-SNAPSET-CONFIGURATION PUBSET=A[,SNAPSET=*ALL]
```

Vor Restaurierung einer Datei bzw. Jobvariable kann der Benutzer feststellen, welcher Sicherungsstand, d.h. welcher Snapset, verwendet werden soll.

#### Informationen über gesicherte Dateien/Jobvariablen anzeigen

Eine Übersicht über Dateien bzw. Jobvariablen, die in einer Snapset-Sicherung vorhanden und damit aus dieser Sicherung restaurierbar sind, erhält der Benutzer mit dem Kommando LIST-FILE-FROM-SNAPSET bzw. LIST-JV-FROM-SNAPSET. Die Kommandos sind an allen Pubset-Sharern parallel und synchron ausführbar.

Der nichtprivilegierte Benutzer erhält Informationen über Dateien und Jobvariablen, auf die er gemäß DVS-Rechte zugreifen kann (vergleichbar mit SHOW-FILE-ATTRIBUTES bzw. SHOW-JV-ATTRIBUTES). Die Systembetreuung hat entsprechend die erweiterten Rechte.

Zur Selektion einer Menge von Dateien/Jobvariablen ist Teilqualifizierung und die Angabe von Musterzeichen im Datei-/Jobvariablennamen, nicht aber in der Katalog- und/oder Benutzerkennung erlaubt.

Die ausgegebenen Informationen enthalten neben den Namen noch die relevanten Attribute zur Restaurierbarkeit: Größe, Sonderkennzeichen für migrierte Dateien und Banddateien, Erstellungdatum und Datum des letzten Zugriffs (bzw. Ablaufdatum bei Jobvariablen), sowie den Dateistatus. Der Dateistatus zeigt an, ob die Datei zum Zeitpunkt der Sicherung schreibgeöffnet war (STATE=OPENED bzw. CLOSED). Bei Dateien, die nicht restaurierbar sind (z.B. Dateien mit der Sicherungshäufigkeit BACKUP-CLASS=E oder spezielle Systemdateien), wird STATE=NOREST angezeigt.

#### Dateien/Jobvariablen restaurieren

Dateien bzw. Jobvariablen kann der Benutzer mit dem Kommando RESTORE-FILE-FROM-SNAPSET bzw. RESTORE-JV-FROM-SNAPSET von einem Snapset restaurieren. Die Kommandos sind an allen Pubset-Sharern parallel und synchron ausführbar.

Der nichtprivilegierte Benutzer kann auf Dateien und Jobvariablen gemäß seiner DVS-Rechte zugreifen (Leserechte für die Originaldatei und Eigentümerrechte für die Zieldatei). Die Systembetreuung hat entsprechend die erweiterten Rechte.

Zur Selektion einer Menge von Dateien/Jobvariablen ist Teilqualifizierung und die Angabe von Musterzeichen im Datei-/Jobvariablennamen, nicht aber in der Katalog- und/oder Benutzerkennung erlaubt.

Die angegebenen Dateien/Jobvariablen werden von einem bestimmten Snapset (mit dem Operanden SNAPSET=*LATEST ist der jüngste Snapset voreingestellt) restauriert.
Bei der Restaurierung können aber auch ein Snapset-Bereich (Operand SNAPSET= *INTERVAL) oder alle Snapsets (Operand SNAPSET=*ALL) einbezogen werden. In diesem Fall werden die Dateien/Jobvariablen auf den jeweils neuesten Stand dieser Snapsets hergestellt.

Dateien und Jobvariablen werden, wie beim HSMS-Restore aus Backup-Archiven, mit ihren ursprünglichen Katalogattributen wiederhergestellt. Insbesondere entsprechen der Erstellungs- und Änderungszeitpunkt sowie die Schutzattribute dem Original zum Sicherungszeitpunkt, d.h. zum Zeitpunkt der Snapset-Erstellung.
In einem Protokoll können wahlweise nur die wegen Fehler nicht restaurierten Dateien/Jobvariablen oder auch alle restaurierten Dateien/Jobvariablen aufgelistet werden.

*Zusätzliche Optionen beim Restaurieren*

Beim Erzeugen eines Snapsets werden alle Dateien und Jobvariablen des Pubsets gleichzeitig gesichert. Ebenso werden alle Seiten einer Datei gleichzeitig gesichert. Dadurch sind schreibgeöffnete Dateien crash-consistent gesichert. Diese Dateien werden standardmäßig nicht restauriert (außer Dateien mit dem Indikator ONLINE-SAVE). Die Restaurierung solcher Dateien kann explizit angefordert werden (Operand RESTORE-OPEN-FILES=*YES). Im Falle von ISAM-Dateien kann eine Nachbearbeitung mit dem Kommando REPAIR-DISK-FILE notwendig sein. Dauernd geöffnete Systemdateien für SRPM, GUARDS, und GCF werden bei der Snapset-Erzeugung in einen konsistenten Zustand gebracht, um später eine ordentliche Restauration zu ermöglichen.

Dateien und Jobvariablen können unter einem neuen Namen restauriert werden (Operand NEW-FILE-NAME bzw. NEW-JV-NAME). Die Umbenennung erfolgt entweder durch Angabe einer anderen Benutzerkennung oder eines Dateinnamenspräfix. Der nichtprivilegierte Benutzer kann eine andere Benutzerkennung angeben, wenn er Miteigentümer dieser Kennung ist. Die eigene Benutzerkennung kann er angeben, wenn er eine fremde Datei (Lesezugriff ist erforderlich) auf der eigenen Kennung restaurieren will.

Bereits bestehende Dateien und Jobvariablen werden standardmäßig nicht überschrieben, also nicht restauriert. Das Restaurieren (Überschreiben) dieser Dateien/Jobvariablen kann explizit angefordert werden (Operand REPLACE=*YES). Das Überschreiben erfordert die Berechtigung zum Löschen (z.B. Schreibkennwort). Der privilegierte Aufrufer kann den Schreibschutz explizit ignorieren (Operand IGNORE-PROTECTION=*YES).

*Einschränkungen*

Einzelne Dateigenerationen können nicht restauriert werden, nur ganze Dateigenerationsgruppen.

Für migrierte Dateien und Banddateien werden nur die Katalogeinträge wiederhergestellt. Eine Umbenennung ist in diesem Fall nicht möglich (Ausnahme: Der privilegierte Benutzer unter TSOS kann bei einer Banddatei eine andere Benutzerkennung angeben).

*PLAM-Bibliothekselemente restaurieren*

In LMS kann mit der Anweisung OPEN-LIBRARY eine PLAM-Bibliothek auf einem Snapset im Lesemodus geöffnet werden. Elemente dieser Bibliothek lassen sich dann über die entsprechenden LMS-Anweisungen angezeigen, kopieren oder selektieren.

*Sicherung vom Snapset in ein Backup-Archiv*

Mit der HSMS-Anweisung BACKUP-FILES können Dateien/Jobvariablen, die auf einen Snapset gesichert wurden, in ein Backup-Archiv übernommen werden. Mit der folgenden BACKUP-Anweisung können z.B. alle Dateien und Jobvariablen der Benutzerkennung `USER1`, die in der letzten Snapset-Sicherung des Pubsets `A` gesichert wurden, in ein Backup-Archiv übernommen werden:

```
//BACKUP-FILES FILE-NAMES=$USER1., JV-NAMES=$USER1.,
               ARCHIV-NAME=<archiv>,CONCURRENT-COPY=*YES(
                WORK-FILE-NAME=*FROM-SNAPSET(PUBSET-ID=A,SNAPSET-ID=-1)
```

Diese Funktion steht auch nichtprivilegierten Benutzern zur Verfügung.

Die Übernahme vom Snapset in ein Backup-Archiv ist zu jedem beliebigen Zeitpunkt zwischen Erzeugen und Löschen des Snapsets möglich. Sie erfolgt praktisch „offline“ zum laufenden Pubset-Betrieb. Die in HSMS neu erzeugte Sicherungsversion erhält als Sicherungszeitpunkt dann nicht das aktuelle Datum des BACKUP-FILES-Aufrufs, sondern das Erstellungsdatum des Snapsets. Im Backup-Archiv darf jedoch noch keine neuere Sicherungsversion existieren.

### 13.2.4 Restaurierung von Pubsets

Die Systembetreuung kann mit dem Kommando RESTORE-PUBSET-FROM-SNAPSET auch den gesamten Pubset auf Volume-Basis aus dem jüngsten Snapset restaurieren.



Für Speichersysteme ETERNUS DX wird diese Funktion derzeit nicht unterstützt und der Kommandoaufruf wird mit der Meldung NDE1811 abgewiesen.

Diese Funktion ist nicht im laufenden Pubset-Betrieb aufrufbar. Der Pubset muss exportiert sein, d.h. er darf weder lokal noch remote in einem anderen System importiert sein. Außerdem muss der Pubset noch auf den Originalplatten existieren und die ursprünglichen Pubset-Eigenschaften aufweisen (siehe auch „Änderung der Pubset-Konfiguration“ auf Seite 548).

Beim Restaurieren wird der Pubset im Plattenspeichersystem auf den Stand des letzten vorhandenen Snapsets gebracht. Dies geschieht praktisch sofort und ohne Ein/Ausgaben von Systemen. Dieser Snapset wird nach dem Kopiervorgang gelöscht und der Pubset besitzt dann den Stand wie unmittelbar vor dem Erzeugen des letzten Snapsets.

Der Pubset bleibt bis zur vollständigen Restaurierung gesperrt. Bis zum Abschluss der Restaurierung ist er gegen die Inbetriebnahme durch /IMPORT-PUBSET geschützt (im SVL ist der PVSREN-Indikator gesetzt). In dieser Zeit werden Import-Versuche mit der Meldung DMS0351 und Insert 03 abgewiesen. Falls bereits die Daten von den Snap-Units zurückkopiert werden, erfolgt die Abweisung mit den Meldungen DMS0381 und DMS038F.

Während der Restaurierung wird auf dem Home-Pubset eine temporäre Kopie des Snapset-Katalogs des zu restaurierenden Pubsets mit dem Dateinamen $TSOS.SYSWRK.SNAPSET.<catid> angelegt. Diese Datei wird nur für die Restaurierung benötigt und erst nach Abschluss der Restaurierung wieder gelöscht. Während der Restaurierung ist sie gegen Löschen geschützt, da sie zum Wiederaufsetzen einer unterbrochenen Restaurierung dient (s.u.). Wenn der Pubset auf eine andere Weise restauriert wird, so wird die Datei beim nächsten Pubset-Import gelöscht.

Nach einem Abbruch des Systemlaufs während der Restaurierung befindet sich der Pubset in einem Zwischenzustand, der einer teilweisen Restaurierung entspricht. In diesem Fall kann die Restaurierung durch einen erneuten Kommandoaufruf beendet werden. Die Restaurierung kann jedoch nur in der gleichen Ablaufumgebung wieder aufgesetzt werden, d.h. mit dem gleichen Home-Pubset, da sich dort die erforderliche Kopie des Snapset-Katalogs befindet.

Um den Pubset auf den Stand eines älteren Snapsets zurückzusetzen, muss die Funktion entsprechend wiederholt werden.

Beim ersten Importieren nach einem Pubset-Restore erfolgt eine Rekonstruktion des F5-Labels.

*Änderung der Pubset-Konfiguration*

Wenn der Pubset nach der letzten Snapset-Sicherung um Platten reduziert wurde und wenn diese Platten mit VOLIN neu initialisiert wurden, dann ist die Restaurierung des Pubsets nicht mehr möglich. Das Kommando RESTORE-PUBSET-FROM-SNAPSET wird in diesem Fall abgewiesen.

Die Restaurierbarkeit des Pubsets nach einer Pubset-Reduzierung kann mit einer der folgenden Maßnahmen erhalten werden:

- weggefallene Platten nicht neu initialisieren
- unmittelbar einen neuen Snapset erzeugen



VOLIN erlaubt keine Initialisierung von Snap-Units und reagiert bei der Initialisierung von Normalplatten mit einer Rückfrage, wenn noch Snap-Units verkettet sind.

## 13.3 Rekonstruktion von Dateien und Datenträgern

Daten können sowohl auf Datei- als auch auf Datenträgerebene gesichert und rekonstruiert werden. Es gibt mehrere Gründe, die eine Rekonstruktion von Dateien bzw. Platten erforderlich machen:

- Eine oder mehrere Dateien müssen aus den Sicherungsbeständen eingespielt werden, weil sie aus Kapazitätsgründen nicht online gehalten werden konnten.
- Das gesamte System muss nach einem Ausfall rekonstruiert werden.
- Eine Platte muss rekonstruiert werden.

Das Ziel der Rekonstruktion ist die schnelle Bereitstellung der gesicherten Dateien.
Je nachdem, ob es sich bei der ausgefallenen Platte um eine gemeinschaftliche oder private Platte handelt, sind unterschiedliche Maßnahmen zu ergreifen.

#### Rekonstruktion einzelner Dateien

Verliert ein Benutzer eine Datei durch fehlerhafte Bedienung – wie z.B. durch versehentliches Löschen – gibt es verschiedene Möglichkeiten der Rekonstruktion durch Rückgriff auf die letzte Sicherungskopie.

- Rückgriff auf die logische Sicherung der Datei

  Die Datei ist logisch auf einem privaten Datenträger gesichert. Die Sicherungskopie ist zu übernehmen und die inzwischen an der Originaldatei vorgenommenen Änderungen sind nachzutragen.

- Rückgriff auf die physikalische Sicherung der Platte oder des Bandes, auf der/dem die Datei liegt (Ausnahmefall)

  Die letzte physikalische Sicherung mit FDDRL liegt auf einer Sicherungsplatte vor oder muss vom Sicherungsband auf eine Platte eingespielt werden. Die betreffende Datei wird von dieser Platte auf die Originalplatte kopiert. Die inzwischen vorgenommenen Änderungen in der ersetzten Datei sind in der Sicherungskopie nachzutragen.

### Rekonstruktion einer gemeinschaftlichen Platte

Nach dem Ausfall einer gemeinschaftlichen Platte muss zunächst dafür gesorgt werden, dass eine funktionsfähige Platte, die auf einem Reservegerät montiert wird, zur Verfügung steht.

Bei Ausfall einer gemeinschaftlichen Platte muss der gesamte Pubset rekonstruiert werden. Der Rekonstruktionsaufwand hängt vom Umfang und der Aktualität der zur Verfügung stehenden letzten physikalischen Sicherung ab.
Diese Sicherung ist zunächst vom Band einzuspielen. Je nach Stand dieser physikalischen Sicherung müssen noch logische Sicherungen eingelesen werden, um den letzten aktuellen Stand zu erhalten.

Danach sind die Katalogeinträge der Benutzerdateien, die auf privaten Platten liegen, zu importieren oder die letzte logische Sicherung einzuspielen. Dies kann mit einer Kommandofolge erfolgen, die ein IMPORT-FILE-Kommando für jede montierte private Platte enthält.

### Rekonstruktion eines SM-Pubsets

Ist in einem SM-Pubset der Control-Volume-Set nicht mehr zugreifbar, fällt der gesamte SM-Pubset aus.

Um die Daten auf den anderen Volume-Sets weiterverarbeiten zu können, kann der exklusive Import eines Volume-Sets erfolgen, der dieses zu einem SF-Pubset mit gleicher Katalogkennung konvertiert. Dazu wird im Kommando IMPORT-PUBSET der Operand `USE=*EXCLUSIVE (CONVERT-VOLUME-SET=*YES)` zur Verfügung gestellt.

Control-Volume-Sets sind von einer Konvertierung ausgeschlossen.

Die Daten und Metadaten des ausgefallenen Control-Volume-Sets müssen nach Bereitstellung eines zusätzlichen SF-Pubsets aus der Sicherung rekonstruiert werden. Danach kann mit dem Dienstprogramm SMPGEN aus den konvertierten SF-Pubsets und dem ebenfalls als SF-Pubset vorliegenden rekonstruierten Control-Volume-Set ein neuer SM-Pubset erzeugt werden.

### Rekonstruktion eines Pubsets

Wenn ein Pubset (SM oder SF), der im Snapset-Betrieb gesichert wurde, nicht mehr importiert werden kann, lässt sich der Pubset in Abhängigkeit vom Plattenspeichersystem mit dem Kommando RESTORE-PUBSET-FROM-SNAPSET auf dem Stand des jüngsten Snapsets restaurieren (siehe Abschnitt „Restaurierung von Pubsets“ auf Seite 553).

### Rekonstruktion einer privaten Platte

Nach Bereitstellen eines intakten Plattengerätes müssen folgende Arbeiten durchgeführt werden:

Die private Platte ist mit dem Dienstprogramm VOLIN zu initialisieren (siehe Handbuch „Dienstprogramme“ [15]). Bei der Rekonstruktion durch FDDRL ist die Sicherung vom Band einzuspielen. Je nach Stand dieser physikalischen Sicherung kann noch das Einlesen logischer Sicherungen nötig sein, um den aktuellen Stand wieder herzustellen.
Im Anschluss daran sind alle Katalogeinträge von Dateien, die im Katalog einen Eintrag für den betreffenden Datenträger hatten, zu löschen (mit dem EXPORT-FILE-Kommando). Danach können die Dateien mit ARCHIVE wieder rekonstruiert werden. Dabei ist zu beachten, dass bei Dateien, die auf mehreren Privatplatten hinterlegt sind, alle betroffenen Datenträger rekonstruiert werden müssen. Ansonsten ist aber jede ARCHIVE-Sicherung logisch, d.h jede Datei kann einzeln rekonstruiert werden. Hier ist empfehlenswert, die Privatplatten gezielt – je Platte eine FILES-Anweisung in ARCHIVE – zu sichern; die Rekonstruktion kann dann über die zugehörige SVID erfolgen. Zudem kann in der FILES-Anweisung der Operand ORIGIN genutzt werden, mit dem die VSN des Datenträgers, dessen Dateien rekonstruiert werden sollen, angegeben wird.

Das folgende Bild zeigt eine Rekonstruktion nach Systemfehler.

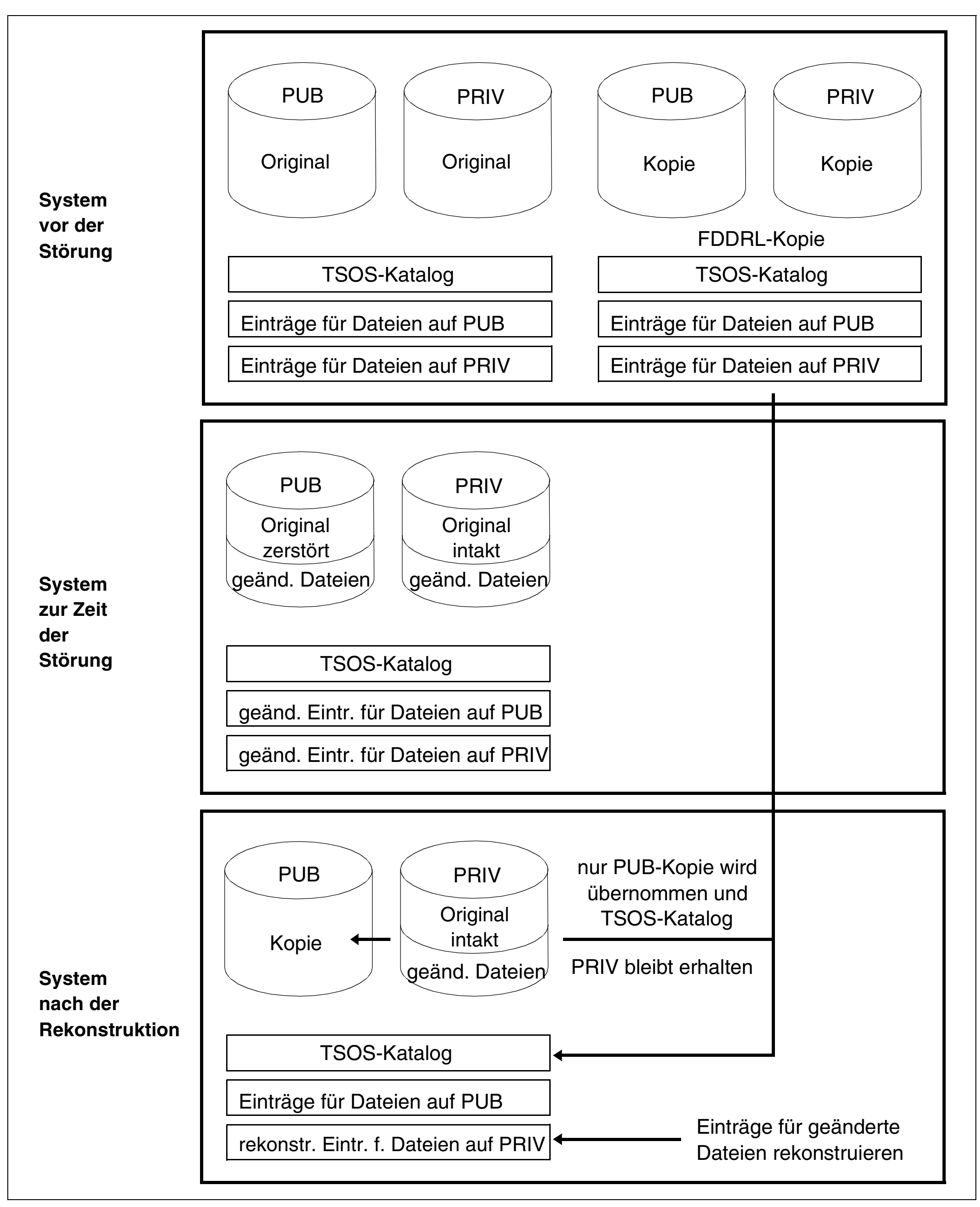



Bild 30: Rekonstruktion nach Systemfehler

## 13.4 Archivsysteme

Die roboterunterstützte Automatisierung der Peripheriebedienung zeichnet sich durch folgende Vorteile aus:

- Verlegung zeitaufwändiger Sicherungsläufe in unbediente Betriebszeiten
- rationelle Nutzung der Kassettenlaufwerke
- Erhöhung der Datensicherheit durch Installation des Archivierungssystems in abgesicherten und zugangskontrollierten Räumen
- Standardisierung der Aktionen und Antworten auf Bandanforderungen
- Entlastung der Plattenperipherie bei vollautomatischem Zugriff auf externe Bestände durch Auslagerung wenig benutzter Online-Daten von Platte auf Kassette

BS2000/OSD unterstützt reale Archivsysteme. Sie werden über das Software-Produkt ROBAR (siehe Handbuch „ROBAR“ [43]) gesteuert.

Die Datenträger können von MAREN verwaltet werden. Die Entscheidung, welches Archivsystem betroffen ist, trifft MAREN anhand des Lagerortes (zu Lagerorten siehe auch Seite 241).

Zu MAREN siehe auch den Abschnitt „Einsatz von MAREN“ auf Seite 249 und die „MAREN“-Handbücher [31].

Neben realen Archivsystemen kann auch das virtuelle Archivsystem **ETERNUS CS HE** zum Einsatz kommen. Mit ETERNUS CS HE wird dem realen Bandrobotersystem (mit den realen Laufwerken und Kassetten) ein virtuelles Bandrobotersystem vorgelagert. Auf diese Weise ist eine völlige Entkopplung zwischen Server und realem Archiv realisiert. Das virtuelle Bandrobotersystem kennt sogenannte virtuelle (logische) Laufwerke und virtuelle (logische) Volumes. Das Kernstück besteht dabei im wesentlichen aus einem Plattensystem als Daten-Cache, so dass nicht nur ein extrem schneller Zugriff auf die Daten gewährleistet wird, sondern auch aufgrund der hohen Anzahl zu generierender virtueller Laufwerke (bis zu 1024) und logischer Volumes (bis zu 1.500.000) bestehende Engpässe bei einem realen Archivsystem beseitigt werden.

Zu ETERNUS CS HE siehe auch das Handbuch „ETERNUS CS HE“ [8].

# 14 Accounting

Das Abrechnungssystem von BS2000/OSD hat die Aufgabe, Daten für die Benutzerabrechnung und die Betriebsabrechnung zu sammeln und bereitzustellen.

Durch die für die Benutzerabrechnung erfassten Daten kann das Data Center den Benutzern die in Anspruch genommenen Betriebsmittel und Dienstleistungen in Rechnung stellen.
Die gesammelten Daten für die Betriebsabrechnung geben eine zeitlich lückenlose Auskunft über die Benutzung und Verfügbarkeit des gesamten Server-Systems.

Die Daten werden in Form von Abrechnungssätzen in eine eigens dafür vorgesehene Abrechnungsdatei geschrieben. Die Abrechnungsdatei wird mit speziellen Programmen ausgewertet.

Am Ende des Kapitels sind alle Abrechnungssätze von BS2000/OSD in einer Übersicht dargestellt (eine ausführliche Beschreibung der Abrechnungssätze und ihrer Struktur finden Sie im Handbuch „Abrechnungssätze“ [1]).

## 14.1 Steuerung des Abrechnungssystems

Das gesamte Abrechnungssystem von BS2000/OSD wird von der Systembetreuung gesteuert. Diese bestimmt den Zeitpunkt, an dem das Abrechnungssystem gestartet werden soll, vereinbart den Namen der Abrechnungsdatei und legt Namen und Anzahl der Abrechnungssätze und Satzerweiterungen fest, die in der Abrechnungsdatei erfasst werden sollen.
Die Systembetreuung legt ferner den Zyklus und den Umfang der periodischen Erfassung fest, die bestimmte Abrechnungsätze und Jobklassen umfasst.

Der festgelegte Ablauf kann mit der Accounting-Exit-Routine derart modifiziert werden, dass der Abrechnungssatz geändert, unterdrückt oder durch weitere Abrechnungssätze ergänzt wird (siehe Handbuch „System-Exits“ [58]). Die Accounting-Exit-Routine wird aktiviert, bevor ein Abrechnungssatz in die Abrechnungsdatei geschrieben wird.

Mit den Kommandos START- und STOP-ACCOUNTING wird das Abrechnungssystem aktiviert bzw. beendet.

Über die Kommandos ADD-USER und MODIFY-USER-ATTRIBUTES kann für jeden Benutzer die Anzahl der benutzerspezifischen Datensätze festgelegt werden, die in die Abrechnungsdatei geschrieben werden darf.

Mit dem Abrechnungssystem kann die Systembetreuung dynamisch Abrechnungssätze ganz oder teilweise ein- und auszuschalten und Einfluss auf den Umfang der einzelnen Abrechnungssätze nehmen.
Mit dem Kommando MODIFY-ACCOUNTING-PARAMETERS können nicht benötigte Abrechnungssätze und Satzerweiterungen ausgeschaltet werden.

Mit dem Kommando SHOW-ACCOUNTING-STATUS informiert sich die Systembetreuung über folgende Daten

- Zustand des Abrechnungssystems (eingeschaltet oder ausgeschaltet)
- Name der aktuellen Abrechnungsdatei
- Namen der Folgedateien
- ein- und ausgeschaltete Sätze bzw. Satzerweiterungen
- Häufigkeit der periodischen Erfassung bestimmter Abrechnungssätze
- Namen der Jobklassen, über die zyklisch Daten erfasst werden

| **Kommando** | **Bedeutung** |
|---|---|
| ADD-USER | Abrechnungsspezifische Benutzerkennungs-Einträge erstellen |
| CHANGE-ACCOUNTING-FILE | Abrechnungsdatei wechseln |
| CREATE-FILE | Vereinbaren der Dateimerkmale der Abrechnungsdatei oder Anlegen der Abrechnungsdatei auf Band oder privater Platte |
| MODIFY-ACCOUNTING-PARAMETERS | Parameter der Systemabrechnung festlegen |
| MODIFY-DEFAULT-ACCOUNT | Standard-Abrechnungsnummern für BS2000- und POSIX-Zugang vereinbaren |
| MODIFY-FILE-ATTRIBUTES | Vereinbaren der Dateimerkmale der Abrechnungsdatei |
| MODIFY-USER-ATTRIBUTES | Abrechnungsspezifische Benutzerkennungs-Einträge ändern |
| SHOW-ACCOUNTING-STATUS | Informationen über das Abrechnungssystem anfordern |
| SHOW-USER-ATTRIBUTES | Abrechnungsspezifische Informationen aus dem Benutzerkatalog anfordern |
| START-ACCOUNTING | Abrechnungssystem aktivieren |
| STOP-ACCOUNTING | Abrechnungssystem beenden |
| WRITE-ACCOUNTING-RECORD | Benutzer-Abrechnungssatz schreiben |
| **Makro** | **Bedeutung** |
| ARDS | DSECT zur Beschreibung der Abrechnungssätze erzeugen |
| AREC | Benutzer-Abrechnungssatz schreiben |
| ASPC | Erfassung der momentanen Speicherplatzbelegung auf gemeinschaftlichen oder privaten Datenträgern in Form von Accounting-Abrechnungssätzen |
| SRMUINF | Daten für die Abrechnung aus dem Benutzerkatalog lesen |

Tabelle 43: Schnittstellenübersicht für die Systemabrechnung

## 14.2 Abrechnungsdatei

Der Dateiname für die Abrechnungsdatei kann bei der Systemeinleitung über den Parameterservice oder während des Systemlaufs festgelegt werden.

- Systemeinleitung

  Das Abrechnungssystem kann im Rahmen der Systemeinleitung aktiviert werden, indem die dazu notwendigen Anweisungen über den Parameterservice mitgeteilt werden. Die Aktivierung des Abrechnungssystems im Rahmen des Parameterservice ist im Abschnitt „Starten des Abrechnungssystems (ACCOUNT)“ auf Seite 82 beschrieben.

- BS2000-Systemlauf

  Die Systembetreuung kann die Abrechnungsroutine im laufenden BS2000/OSD mit dem Kommando START-ACCOUNTING einschalten. Das ist dann sinnvoll, wenn das Abrechnungssystem z.B. zu Testzwecken für eine Weile ausgeschaltet war.
  Soll die aktuelle Abrechnungsdatei ausgewertet werden, wird über das Kommando CHANGE-ACCOUNTING-FILE die aktuelle Abrechnungsdatei geschlossen und eine neue Datei eröffnet.

#### Automatische Generierung des Dateinamens

Die Dateinamen für die Abrechnungsdateien können vom System automatisch generiert werden. Der Dateiname setzt sich aus dem Präfix $TSOS.SYS.ACCOUNT. und einem für alle Loggingverfahren gültigen Suffix yy.mm.dd.xxx.nn bzw. yyyy-mm-dd.xxx.nn zusammen (vgl. Dateiname der Protokolldatei CONSLOG und SERSLOG-Datei).

Dabei bedeutet:

| | |
|---|---|
| yyyy-mm-dd<br>oder<br>yy.mm.dd | Datum, an dem die Abrechnungsdatei eingerichtet wird (ob das Jahr zwei- oder vierstellig ausgegeben wird, ist abhängig vom Systemparameter FMTYFNLG) |
| xxx | Nummer des aktuellen Systemlaufs |
| nn | laufende Nummer der Abrechnungsdatei<br>(01-99; beginnt beim STARTUP immer mit 01) |

Die Systembetreuung kann die Voreinstellung für das Präfix $TSOS.SYS.ACCOUNT ändern, wenn sie bestimmte Namenskonventionen im Data Center berücksichtigen muss.

Wenn an Stelle eines vollqualifizierten Dateinamens ein teilqualifizierter Dateiname angegeben wird, sind folgende Punkte zu beachten:

- Ein teilqualifizierter Dateiname wird automatisch durch das Suffix yy.mm.dd.xxx.nn bzw. yyyy-mm-dd.xxx.nn ergänzt. Aus der Länge des Suffixes ergibt sich ein maximal 26 Zeichen langer teilqualifizierter Dateiname (ohne Benutzerkennung); bei einer Katalogkennung mit mehr als einem Zeichen reduziert sich dieser Wert um die Anzahl der Mehrstellen.
- Wurde der Dateiname ohne Benutzerkennung angegeben, dann wird $TSOS. dem Dateinamen vorangestellt.
- Enthält der teilqualifizierte Dateiname nur die Benutzerkennung, wird der Name erweitert auf $userid.SYS.ACCOUNT.<date>.xxx.nn (<date>= yy.mm.dd oder yyyy-mm-dd).

**Mögliche Fehler beim Eröffnen des Abrechnungssystems**

- Bei Syntaxfehlern in den Parametersätzen während der Systemeinleitung fordert eine Fehler-Routine des Parameterservices den Operator an der Bedienstation auf, den fehlerhaften Parameter zu korrigieren oder zu ignorieren.
- Für die zu eröffnende Datei wird vom Abrechnungssystem ein FILE-Makro abgesetzt. Tritt dabei ein Fehler auf, dann wird eine Abrechnungsdatei mit dem Standardnamen $TSOS.SYS.ACCOUNT.<date>.xxx.nn und den entsprechenden Standardattributen angelegt.
- Dies gilt auch für den Fall, dass die Datei, die eröffnet werden sollte, bereits existiert und mit dem Schutzattribut ACCESS=READ katalogisiert ist.
  Zu beachten ist, dass in vorhergehenden Betriebssystemversionen die Abrechnungsdateien mit ACCESS=READ katalogisiert wurden.
- Nach maximal drei erfolglosen Versuchen, die Datei mit dem Standardnamen zu eröffnen, erscheint eine Meldung auf der Bedienstation (keine Antwort notwendig), und BS2000/OSD wird ohne Abrechnungssystem geladen.
  Die Systembetreuung kann das Abrechnungssystem im aktuellen Systemlauf per Kommando einschalten.

## 14.2.1 Merkmale und Inhalt der Abrechnungsdatei

Die Abrechnungsdatei ist eine SAM-Datei mit variabler Satzlänge. Die übrigen Dateimerkmale kann die Systembetreuung sowohl über die Kommandos CREATE-FILE bzw. MODIFY-FILE-ATTRIBUTES als auch über die Abrechnungskommandos START-ACCOUNTING und CHANGE-ACCOUNTING-FILE vereinbaren.

Folgende Punkte sind zu beachten:

- Als Standardwerte gelten für

  | | |
  |---|---|
  | SPACE: | 48 PAM-Blöcke Primär- und<br>48 PAM-Blöcke Sekundärzuweisung |
  | BUFFER-LENGTH: | 1 PAM-Block bzw. 2 PAM-Blöcke bei NK4-Pubsets |
  | VOLUME: | wird vom DVS festgelegt |

- Wird der VOLUME-Operand bei den Abrechnungskommandos vereinbart, dürfen nur Datenträger vereinbart werden, die vom Datenverwaltungssystem von BS2000/OSD automatisch unterstützt werden und keine zusätzlichen gerätespezifischen Angaben benötigen.
  Soll die Abrechnungsdatei auf **Band** oder **privater Platte** angelegt werden, muss die Datei vor dem Starten des Abrechnungssystems mit dem Kommando CREATE-FILE angelegt werden. Gehört im Falle einer **gemeinschaftlichen Platte** diese nicht dem Standard-Pubset der vorgesehenen Benutzerkennung an, muss die Katalogkennung des Datenträgers im Dateinamen mit angegeben werden.
- Die Abrechnungsdatei wird im EXTEND-Modus eröffnet. Das bedeutet, dass eine vorhandene Datei fortgeschrieben wird.
- Eine neue Abrechnungsdatei wird mit den Schutzattributen ACCESS=WRITE und USER-ACCESS=OWNER-ONLY katalogisiert.
  Existiert die Abrechnungsdatei bereits, dann wird keine Veränderung der Attribute vorgenommen. Die Systembetreuung muss demzufolge darauf achten, dass die Abrechnungsdatei mit den entsprechenden Attributen versehen wird.
- Wird für eine Datei nur ein Katalogeintrag angelegt und noch kein Speicherplatz reserviert (Kommando CREATE-FILE mit Operand SUPPORT=NONE), so tritt beim Versuch, diese Datei als Abrechnungsdatei zu eröffnen, ein Fehler auf.
  Um das zu verhindern, muss die Systembetreuung dieser Datei vor dem Eröffnen primären Speicherplatz zuweisen.

Die Abrechnungsroutine schreibt folgende Informationen in die Abrechnungsdatei:

- Taskabrechnung
- Programmabrechnung
- Geräte- und Datenträgerabrechnung
- Speicherplatzabrechnung
- SPOOLOUT-Abrechnung
- Benutzer-spezifische Abrechnung
- Betriebsabrechnung
- Betriebsmittelverfügbarkeit
- Auftragnehmer-Leistungs-Abrechnung
- Abrechnung entbündelter Produkte
- Abrechnung von Subsystemen und Subsystemtasks

Die Informationen zur Programmabrechnung und zur Speicherplatzabrechnung werden von der Abrechnungsroutine zusätzlich periodisch erfasst. Damit ist eine genauere Abrechnung lange laufender Programme und bestimmter Jobklassen, die dieser periodischen Überwachung unterliegen, möglich.

Die Abrechnungsdaten stehen in der Abrechnungsdatei in einzelnen Abrechnungssätzen. Die Abrechnungssätze haben eine variable Satzlänge; sie beträgt maximal 492 Byte ohne Satzlängenfeld und 496 Byte mit Satzlängenfeld.

Die Struktur eines Abrechnungssatzes ist ausführlich im Handbuch „Abrechnungssätze“ [1] beschrieben.

## 14.2.2 Arten von Abrechnungsdaten

### Programmabrechnung mit Verbrauchsstempeln

Der Betriebsmittelverbrauch eines Programmlaufs wird in zwei Abrechnungssätzen beim Start und bei der Beendigung des Programms erfasst, die den jeweils aktuellen Stand der Verbrauchswerte (*Verbrauchsstempel*) seit Beginn des Tasklaufs enthalten.
Ein Tasklauf wird vom Abrechnungssystem in die Ereignisse Task-Start, Programm-Start, Programm-Beendigung und Task-Beendigung aufgeteilt. Ein Auswertungsprogramm kann durch Differenzbildung der zugehörigen Ereignisdaten sowohl den Betriebsmittelverbrauch des Programms als auch die Verbrauchsdaten des zwischen zwei Programmen liegenden Kommando-Modus berechnen. Zur genaueren Abrechnung lange laufender Programme bietet das Abrechnungssystem auch einen dritten Satz an, der in periodischen Abständen Daten in die Abrechnungsdatei schreibt und Aufschluss über die durchschnittlichen Verbrauchswerte während der Laufzeit gibt.

### Abrechnungsdaten für die Benutzerabrechnung

Für die Benutzerabrechnung stellt das Abrechnungssystem eine Reihe von Abrechnungsdaten zur Verfügung, die im Einzelnen der genauen Beschreibung der Abrechnungssätze im Anhang zu entnehmen sind. Folgende Daten werden für eine Benutzerkennung bereitgestellt:

- Daten zu den Basis-Betriebsmitteln
  - CPU-Zeit
  - E/A-System (Anzahl der transportierten Blöcke oder Byte)
  - Hauptspeicher (Working-Set-Integral)
  - Service Units (Summe aus den Einzelwerten CPU-Zeit, Ein-/Ausgaben und Hauptspeicher, berücksichtigt die Prozessorleistung)
- Erweiterte Gerätebelegungsdaten mit Zeitstempel und Gerätetyp
- Belegungsdaten des Datenträgers
- Job-Informationen und Task-Ablauf-Informationen
- Erweiterte Programm-Informationen
  - Interner Programmname
  - Beendigungsgrund
  - Speicherbelegungsintegral für Klasse-5- und Klasse-6-Speicher
- Spoolout-Informationen
  - Typ des Ausgabegerätes
  - Anzahl gedruckter Zeilen und Seiten
- Abrechnung von Speicherplatz-Änderungen

### Speicherplatz-Abrechnung

Für die Abrechnung der permanenten Speicherplatz-Belegung auf gemeinschaftlichen und privaten Platten bietet das Abrechnungssystem eine Schnittstelle an, die eine Erfassung über Stichproben ermöglicht. Diese Stichproben können dabei von einem Benutzerprogramm (über Makro ASPC) unter der privilegierten Benutzerkennung der Systembetreuung gesteuert werden.
Der Makro ASPC schreibt dabei für jeden lokal verfügbaren Pubset einen oder mehrere Abrechnungssätze in die Abrechnungsdatei, wobei jeder Satz folgende Daten enthält:

- die Katalogkennung des Pubsets
- den Zeitpunkt der Bestandsaufnahme
- einen Indikator über die Vollständigkeit des Satzes
- die Benutzerkennungen des Pubsets
- die Anzahl der belegten PAM-Blöcke pro Benutzerkennung

Der Makro ASPC ist im Handbuch „Makroaufrufe an den Ablaufteil“ [30] beschrieben.

Diese Datensätze können auch periodisch in der Abrechnungsdatei erfasst werden. Die Parameter für diese zyklische Überwachung und Erhebung steuert die Systembetreuung mit entprechenden Vorgaben auf Kommandoebene.

### Betriebsabrechnung

Die Verfügbarkeit des Gesamtsystems wird durch einen Abrechnungs-Eröffnungs- und einen Abrechnungs-Abschlusssatz dokumentiert. Der Eröffnungssatz wird nach jeder Eröffnung einer Abrechnungsdatei geschrieben, der Abschlusssatz vor jedem ordnungsgemäßen Schließen der Abrechnungsdatei.
Diese Sätze enthalten im Kennzeichnungsteil bzw. in der Grundinformation:

- die Bezeichnung des Servers
- den Systemnamen und die Systemversion
- ein Kennzeichen für Startup oder Shutdown

Bei einem Wechsel der Abrechnungsdatei erhält die neue Datei einen Verweis auf ihre Vorgängerin.
Die Verfügbarkeit des Betriebsmittels CPU wird in periodisch aufgezeichneten Sätzen erfasst, die die Task-Zeiten (TU + TPR), Interrupt-Handling-Zeiten (SIH) und IDLE-Zeiten der CPU enthalten. Die Hauptspeichergröße wird im ersten Abrechnungs-Eröffnungssatz während der Systemeinleitung oder nach einem START-ACCOUNTING-Kommando aufgezeichnet.

### Abrechnung von Systemdiensten in Auftragnehmertasks

Die Task ist grundsätzlich die größte Einheit, die für die Benutzerabrechnung herangezogen wird.

Systemfunktionen, die in einer Systemtask oder an einem anderen Server ausgeführt werden und einen nennenswerten Verbrauch an Betriebsmitteln aufweisen, werden in eigenen Abrechnungssätzen erfasst.

Folgende vom Benutzer angeforderte Leistungen werden in entsprechenden Leistungsgrößen ermittelt:

- Spoolout
- Userdumps
- RFA-Verbindungen

Für Systemfunktionen, bei denen der Betriebsmittelverbrauch für einen einzelnen Auftrag verhältnismäßig gering ist, die aber häufig aufgerufen werden, sind die Leistungsgrößen in den Einträgen der Task- und Programm-Abrechnungsätze integriert, z.B.:

- Anzahl Platten-Ein-/Ausgaben
- Anzahl Terminal-Ein-/Ausgaben
- Anzahl Katalogzugriffe

Die Abrechnungssätze sind in den Tabellen ab Seite 572 dargestellt.

## 14.3 Übersichten über die Abrechnungssätze

Die Abrechnungssätze sind einmal nach Zielgruppe (siehe unten) und einmal alphabetisch nach Gruppenzusammengehörigkeit (siehe folgende Seiten) geordnet.

| **Abrechnungssatz** | | | **max. Länge in Byte** |
|---|---|---|---|
| für die Benutzerabrechnung | | | |
| | JOBS | Job-Start-Abrechnungssatz | 314 |
| | PACC | Periodischer Programm-Abrechnungssatz | 402 |
| | PDMP | Userdump-Abrechnungssatz | 78 |
| | PRGS | Programm-Start-Abrechnungssatz | 476 |
| | PRGT | Programm-Beendigungs-Abrechnungssatz | 528 |
| | SPLO | SPOOLOUT-Abrechnungssatz | 474 |
| | TASK | Task-Abrechnungssatz | 400 |
| | TDEV | Geräte- und Datenträger-Abrechnungssatz | 492 |
| | TATR | Task-Attribut-Änderungs-Abrechnungssatz | 70 |
| für die Speicherplatz-Abrechnung | | | |
| | DALC | Speicherplatz-Allokierungs-Abrechnungssatz | 492 |
| | DSPC | Speicherplatz-Bestandsaufnahme-Abrechnungssatz | 492 |
| | DSPP | Privatplatten-Speicherplatz-Bestandsaufnahme-Abrechnungssatz | 492 |
| benutzerspezifische Abrechnungssätze | | | |
| | UACC | Benutzer-Verbrauchsstempel-Abrechnungssatz | 382 |
| | UDAT | Benutzerdaten-Abrechnungssatz | 326 |
| | frei definierte Benutzer-Abrechnungssätze | | 496 (mit SLF) |
| für die Betriebsabrechnung | | | |
| | ACLS | Abrechnungs-Abschluss-Satz | 610 (mit PUT) |
| | AOPN | Abrechnungs-Eröffnungs-Satz | 698 (mit PUT) |
| | RCPU | CPU-Verfügbarkeits-Abrechnungssatz | 70 |
| | RSRV | Auftragnehmer-Leistungs-Abrechnungssatz | [1] |
| für DSSM | | | |
| | ESMC | Subsystem-Initialisierungs-Abrechnungssatz | 54 |
| | ESMD | Subsystem-Beendigungs-Abrechnungssatz | 54 |

Tabelle 44: Übersicht über die Abrechnungssätze von BS2000/OSD

[1] Die Satzlänge hängt ab von der Auftragnehmer-Art

In der folgenden Tabelle bedeutet in der Spalte „Standardeinstellung“: IE/IA = implizit eingeschaltet/implizit ausgeschaltet, EE/EA = explizit eingeschaltet/explizit ausgeschaltet.

| Gruppe | Satz-kennung | Erweite-rungs-kennung | Bedeutung | Aus-schalt-barkeit | Std.-einstel-lung |
|---|---|---|---|---|---|
| A | | | **Abrechnungsdaten des Abrechnungssystems** | | |
| | ACLS | | Abrechnungs-Abschluss-Satz | Nein | IE |
| | | FN | Name der Nachfolgerdatei | Nein | IE |
| | AOPN | | Abrechnungs-Eröffnungssatz | Nein | IE |
| | | FN | Name der Vorgängerdatei | Nein | IE |
| | | MM | Hauptspeicherausbau und Arbeitsspeichergröße | Nein | IE |
| C | | | - reserviert - | | |
| D | | | **Abrechnungsdaten des DVS** | | |
| | DALC | | Speicherplatz-Allokierungs-Satz | Ja | EA |
| | | AL | Speicherplatz-Änderungsdaten | Nein | IE |
| | DRFA [1] | | RFA-Session-Abrechnungssatz | Ja | EA |
| | | ST | RFA-Session-Beendigungsgrund | Nein | IE |
| | DRVR [1] | | Abrechnungsdaten von DRV | Ja | EA |
| | DSPC | | Speicherplatz-Bestandsaufnahme-Satz | Ja | IE |
| | | SP | Speicherplatz-Belegungsdaten | Nein | IE |
| | DSPP | | Speicherplatz-Bestandsaufnahme-Satz für Privatplatten | Ja | IE |
| | | PS | Speicherplatz-Belegungsdaten für Privatplatten | Nein | IE |
| E | | | **Abrechnungssätze von DSSM** | | |
| | ESMC | | Subsystem-Initialisierungs-Abrechnungssatz | Ja | IE |
| | ESMD | | Subsystem-Beendigungs-Abrechnungssatz | Ja | IE |
| F | | | **Abrechnungsdaten von File-Transfer** | | |
| | FTRO [1] | | | Ja | IE |
| | | FN | Name der übertragenen Datei | Ja | IE |
| | | MN | Name der Bibliothek | Ja | IE |
| | | YY | Jahrhundertanteil bei Zeitangaben | Ja | IE |
| | | MS | Rechenzeit-Verbrauch | Ja | IE |
| G | | | - reserviert - | | |
| H | | | **Abrechnungsdaten des Hierarchischen Speicher-Management-Systems** | | |
| | HSMS [1] | | HSMS-Abrechnungssatz | Ja | EA |
| | | IO | Ein-/Ausgaben pro Gerätegruppe | Ja | EA |
| I | | | - reserviert - | | |

| Gruppe | Satz-ken-nung | Erweite-rungs-kennung | Bedeutung | Aus-schalt-barkeit | Std.-einstel-lung |
|---|---|---|---|---|---|
| J | | | **Abrechnungsdaten der Auftragsverwaltung** | | |
| | JOBS | | Job-Start-Abrechnungssatz | Ja | IE |
| | | JO | Job-Herkunft | Ja | IE |
| | | JD | Job-Eigenschaften | Ja | EA |
| | | JR | Betriebsmittel-Anforderung | Ja | EA |
| | | JP | Job-Parameter | Ja | EA |
| K | | | - reserviert - | | |
| L | | | - reserviert - | | |
| M | | | - reserviert - | | |
| N | | | - reserviert - | | |
| O | | | - reserviert - | | |
| P | | | **Programm-Abrechnungsdaten** | | |
| | PACC | | Periodischer Programm-Abrechnungssatz | Ja | EA |
| | | PD | Zeitpunkt des vorhergehenden PACC-Satzes | Ja | IE |
| | | MA | Hintergrundspeicherbelegung | Ja | IE |
| | | IO | E/A und Datenmenge je Gerätegruppe | Ja | IE |
| | | TI | Terminal-Ein-/Ausgaben | Ja | IE |
| | | CA | Anzahl der Katalogzugriffe | Ja | IE |
| | | PC | Performance-Controller-Daten | Ja | IE |
| | | ID | Benutzer-Kennzeichnung | Ja | IE |
| | PDMP | | Userdump-Abrechnungssatz | Ja | IE |
| | PRGS | | Programm-Start-Abrechnungssatz | Ja | IE |
| | | PN | Interner Programmname | Ja | IE |
| | | MA | Hintergrundspeicherbelegung | Ja | IE |
| | | IO | E/A und Datenmenge je Gerätegruppe | Ja | IE |
| | | TI | Terminal-Ein-/Ausgaben | Ja | EA |
| | | CA | Anzahl der Katalogzugriffe | Ja | EA |
| | | PC | Performance-Controller-Daten | Ja | EA |
| | | ID | Benutzer-Kennzeichnung | Ja | EA |
| | PRGT | | Programm-Beendigungs-Abrechnungssatz | Ja | IE |
| | | PT | Programm-Beendigungs-Grund | Ja | IE |
| | | MA | Hintergrundspeicherbelegung | Ja | IE |
| | | IO | E/A und Datenmenge je Gerätegruppe | Ja | IE |
| | | TI | Terminal-Ein-/Ausgaben | Ja | EA |
| | | CA | Anzahl der Katalogzugriffe | Ja | EA |
| | | PC | Performance-Controller-Daten | Ja | EA |
| | | EI | Externe Programm-Identifikation | Nein | IE |
| | | ID | Benutzer-Kennzeichnung | Ja | EA |
| Q | | | - reserviert - | | |

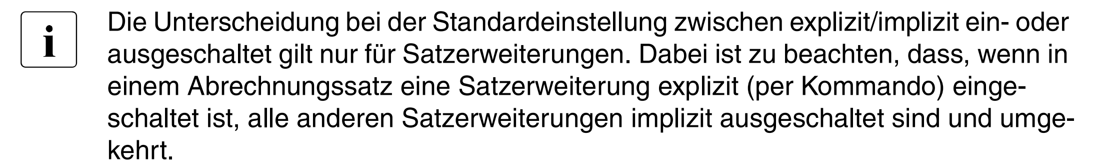
| Gruppe | Satz-kennung | Erweite-rungs-kennung | Bedeutung | Aus-schalt-barkeit | Std.-einstel-lung |
|---|---|---|---|---|---|
| R | | | **Betriebsmittel-Verfügbarkeitsdaten** | | |
| | RCPU | | CPU-Verfügbarkeits-Abrechnungssatz | Ja | EA |
| | RSRV | | Auftragnehmer-Leistungs-Abrechnungssatz | Ja | EA |
| | | RD | Betriebsmittel-Beschreibung | Ja | IE |
| | | SV | Leistungsdaten | Ja | IE |
| S | | | **SPOOL-Abrechnungsdaten** | | |
| | SPLO | | SPOOLOUT-Abrechnungssatz | Ja | IE |
| | | OT | SPOOLOUT-Beendigungsgrund | Ja | IE |
| | | OC | SPOOLOUT-Erzeugung | Ja | IE |
| | | OI | SPOOLOUT-Anstoß | Ja | IE |
| | | IN | SPOOLOUT-Eingabeband | Ja | IE |
| | | OM | Ausgabemedium | Ja | IE |
| | | FN | SPOOLOUT-Dateiname | Ja | IE |
| T | | | **Task-Abrechnungdaten** | | |
| | TASK | | Task-Abrechnungssatz | Nein | IE |
| | | TT | Task-Beendigungs-Grund | Ja | IE |
| | | MA | Hintergrundspeicherbelegung | Ja | IE |
| | | IO | E/A und Datenmenge je Gerätegruppe | Ja | IE |
| | | TI | Terminal-Ein-/Ausgaben | Ja | EA |
| | | CA | Anzahl der Katalogzugriffe | Ja | EA |
| | | PC | Performance-Controller-Daten | Ja | EA |
| | | ID | Benutzer-Kennzeichnung | Ja | EA |
| | TATR | | Task-Attribut-Änderungs-Abrechnungssatz | Ja | EA |
| | TDEV | | Geräte- und Datenträger-Belegungssatz | [2] | IE |
| | | DU | Unit-Record-Geräte-Belegungsdaten | Ja | IE |
| | | DV | Volume-Geräte-Belegungsdaten | Ja | IE |
| | | VU | Datenträger-Belegungsdaten | Ja | EA |
| | | ID | Benutzer-Kennzeichnung | Ja | EA |

| Gruppe | Satz-kennung | Erweite-rungs-kennung | Bedeutung | Aus-schalt-barkeit | Std.-einstel-lung |
|---|---|---|---|---|---|
| U | | | **Unprivilegierte Benutzer-Abrechnungsdaten und UTM-Abrechnungsdaten** | | |
| | UACC | | Benutzer-Verbrauchsstempel-Satz | Ja | IE |
| | | ID | Benutzer-Kennzeichnung | Nein | EE |
| | | MA | Hintergrundspeicherbelegung | Ja | IA |
| | | IO | E/A und Datenmenge je Gerätegruppe | Ja | IA |
| | | TI | Terminal-Ein-/Ausgaben | Ja | IA |
| | | CA | Anzahl der Katalogzugriffe | Ja | IA |
| | | PC | Performance-Controller-Daten | Ja | IA |
| | UDAT | | Benutzerdaten-Satz | Ja | IE |
| | | [3] | Benutzerdaten-String | Nein | IE |
| | UTMA [1] | | UTM-Abrechnungssatz | Ja | IE |
| | UTMK [1] | | UTM-Kalkulationssatz | Ja | IE |
| V | | | **Abrechnungsdaten für virtuelle Maschinen** | | |
| | VACD [1] | | VM2000-Abrechnungssatz für Geräte | Ja | IE |
| | | DV | Geräteerweiterung | Nein | IE |
| | VACM [1] | | VM2000-Abrechnungssatz für Betriebsmittel | Ja | IE |
| W | - reserviert - | | | | |
| X | - reserviert für Benutzer-Abrechnungssätze | | | | |
| Y | - reserviert für Benutzer-Abrechnungssätze | | | | |
| Z | - reserviert für Benutzer-Abrechnungssätze | | | | |

[1] Diese Abrechnungssätze sind in den jeweiligen Handbüchern der Produkte beschrieben

[2] Wenn alle Satzerweiterungen dieser Abrechnungssätze ausgeschaltet sind, ist die Satzart implizit ausgeschaltet

[3] Erweiterungskennung: zwei Leerzeichen



Die Unterscheidung bei der Standardeinstellung zwischen explizit/implizit ein- oder ausgeschaltet gilt nur für Satzerweiterungen. Dabei ist zu beachten, dass, wenn in einem Abrechnungssatz eine Satzerweiterung explizit (per Kommando) eingeschaltet ist, alle anderen Satzerweiterungen implizit ausgeschaltet sind und umgekehrt.

Die einzelnen Abrechnungssätze sind ausführlich im Handbuch „Abrechnungssätze“ [1] beschrieben.

# 15 Systemmeldungen

Dieses Kapitel gibt eine Einführung zu den Systemmeldungen von BS2000/OSD.

Die BS2000-Systemmeldungen sind ein wesentliches Element der Kommunikation zwischen Anwender und Betriebssystem. Sie informieren den Anwender über Fehlersituationen, die während der Bearbeitung seines Jobs auftreten, über die erfolgreiche Beendigung eines Arbeitsabschnittes oder fordern zu einer Maßnahme/Eingabe auf.
BS2000-Systemmeldungen haben einen definierten Aufbau, sind in Meldungsdateien gespeichert und werden vom BS2000-Meldungssystem verwaltet.

| **Kommando** | **Bedeutung** |
|---|---|
| HELP-MSG-INFORMATION | Text einer Systemmeldung und Erläuterungen zu dieser Meldung anfordern |
| MODIFY-MIP-PARAMETERS | MIP-Parameterdatei erzeugen oder ändern und/oder Meldungsdateien dem System zuschalten oder entziehen |
| MODIFY-MSG-ATTRIBUTES | Sprache der Meldungsausgabe vereinbaren (tasklokal) |
| MODIFY-MSG-FILE-ASSIGNMENT | Meldungsdateien für den Systemlauf festlegen |
| SHOW-MIP-PARAMETERS | Einträge der MIP-Parameterdatei oder Informationen über die Meldungsdateien für den Systemlauf ausgeben |
| SHOW-MSG-FILE-ASSIGNMENT | Informationen über die Meldungsdateien für den Systemlauf anzeigen |
| **Makro** | **Bedeutung** |
| MSG7 / MSG7X | Meldung ausgeben |
| MSGSINIT / MSGSMOD | Meldungsdatei sperren oder dem Meldungssystem hinzufügen |
| MSGSHOW | Informationen über system- oder taskspezifische Meldungen ausgeben |

Tabelle 45: Schnittstellenübersicht zum Meldungswesen

## 15.1 BS2000-Meldungssystem

Kernstück des BS2000-Meldungssystems ist die Systemkomponente MIP (Message Improvement Processing). MIP erhält von einem gerade ablaufenden Modul einer anderen Systemkomponente, eines Subsystems oder eines Benutzerprogramms über den Makro MSG7/MSG7X die Aufforderung, eine bestimmte Meldung zu einem definierten Ausgabeort (Bedienstation, Datensichtstation) zu senden (siehe Handbuch „Makroaufrufe an den Ablaufteil“ [30]) oder in eine S-Variable auszugeben (siehe Abschnitt „Ausgabe von Meldungen in S-Variablen“ auf Seite 600).

Die Systemkomponente MIP übernimmt bei der Meldungsverarbeitung folgende Aufgaben:

1. Über den Meldungsschlüssel und den Kennbuchstaben für die Sprache wird in den zur Verfügung stehenden Meldungsdateien nach der entsprechenden Meldung gesucht. Die Meldung ist als „Meldungsgerüst“ in der Meldungsdatei abgespeichert.
2. Das „Meldungsgerüst“ wird vervollständigt, d.h. im Meldungstext vorhandene Inserts werden durch aktuelle Werte ersetzt. Diese werden MIP über den Makro MSG7/MSG7X mitgeteilt.
3. Die vollständige Meldung wird zum angegebenen Ausgabeort gesendet.

# 15.2 Meldungsdateien

Meldungen sind in Meldungsdateien gespeichert. Diese Dateien sind als ISAM-Dateien organisiert, wobei der Meldungsschlüssel zugleich der ISAM-Schlüssel ist.

Meldungsdateien werden mit dem Dienstprogramm MSGMAKER erstellt und bearbeitet, siehe Handbuch „Dienstprogramme“ [15]. Dies gilt sowohl für Meldungsdateien, die vom Hersteller ausgeliefert werden, als auch kundeneigene Meldungsdateien.

## 15.2.1 Systemmeldungsdateien

Bereits bei der Systemeinleitung können verschiedene Angaben zum Meldungssystem über den Parameterservice vereinbart werden:
Neben der Anzahl der Meldungsdateien, die über die Zugriffsmethode DLAM verarbeitet werden sollen (MSGDLAM) und der Sprache, in der standardmäßig die Meldungsausgabe erfolgen soll (MSGLPRI), legt die Systembetreuung im Parametersatz SYSOPT-CLASS2 vor allem die Anzahl und die Dateinamen der Systemmeldungsdateien fest, die durch den Parameterservice vereinbart werden. Hierbei gilt für die Systemparameter MSGFIL01 und MSGFIL02 folgende Standardeinstellung: MSGFIL01=SYSMES.BS2CP.version bzw. MSGFIL02=SYSMES.EKP.01. Diese Dateien werden bei der Installation mit IMON automatisch eingerichtet.

Die Systembetreuung kann weitere 13 Meldungsdateien vollqualifiziert angeben (MSGFIL03-MSGFIL15).

Es ist möglich, die weiteren Meldungsdateien dynamisch in einer MIP-Parameterdatei zu halten. Diese Parameterdatei kann im Gegensatz zu den im Parametersatz SYSOPT-CLASS2 stehenden Angaben um Meldungsdateien erweitert bzw. reduziert werden.
Die MIP-Parameterdatei wird vom Systembetreuer mit dem Kommando MODIFY-MIP-PARAMETERS angelegt/verwaltet und bekommt standardmäßig den Namen SYSPAR.MIP.`<version>`. Sie wird auf der Kennung TSOS abgelegt und hat folgenden Aufbau:

```
MIPPAR=SYSTEM
* this must be the first line
*
* comment lines begin with an "*"
*
MSGFILE=$.SYSMES.product_1.version
MSGFILE=$.SYSMES.product_2.version
:
```

Die MIP-Parameterdatei kann beliebig viele Vereinbarungen enthalten. Da BS2000/OSD insgesamt nur 255 Meldungsdateien verwalten kann, werden überzählige Angaben ignoriert. Es wird eine entsprechende Warnung an der Bedienstation ausgegeben.

Die MIP-Parameterdatei darf leer sein, muss aber zum Zeitpunkt der Systemeinleitung immer katalogisiert sein. (Ist sie leer, wird ein neuer Dateiname erfragt. Für diesen Fall ist es von Vorteil, eine MIP-Parameterdatei zu haben, in deren erster Zeile „MIPPAR=SYSTEM“ steht.) Eine mit dem Kommando MODIFY-MIP-PARAMETERS eingefügte Meldungsdatei wird an den Beginn der MIP-Parameterdatei geschrieben, gleich nach der Eröffnungszeile und evtl. nachfolgenden Kommentarzeilen. Die MIP-Parameterdatei kann mit dem Kommando SHOW-MIP-PARAMETERS gelesen werden.

Temporär kann mit den Kommandos MODIFY-MSG-FILE-ASSIGNMENT oder MODIFY-MIP-PARAMETERS ...,SCOPE=*TEMPORARY die Menge der Meldungsdateien während der BS2000-Session beeinflusst werden.

Somit hat der Systembetreuer drei Möglichkeiten zur Auswahl, die Meldungsdateien für die BS2000-Session festzulegen:

- zum Startup-Zeitpunkt im Parameterservice mit den Systemparametern
- während Startup, aber nachdem MIP geladen wurde, mit der MIP-Parameterdatei
- während der BS2000-Session mit den Kommandos MODIFY-MIP-PARAMETERS und MODIFY-MSG-FILE-ASSIGNMENT

Bei Beendigung des Systemlaufs werden alle Meldungsdateien geschlossen.

### 15.2.2 Benutzermeldungsdateien

Der nichtprivilegierte Anwender kann für seine Task Meldungsdateien hinzufügen oder entfernen (/MODIFY-MSG-FILE-ASSIGNMENT) und die Sprache der Meldungsausgabe festlegen. Er kann somit auf Fehlersituationen mit eigenen Meldungen reagieren.

Die entsprechenden Kommandos und Makros sind in den Handbüchern „Kommandos“ [27] und „Makroaufrufe an den Ablaufteil“ [30] beschrieben.

## 15.3 Garantierte Meldungen

Meldungsschlüssel und Inserts diverser Systemmeldungen werden für zukünftige Subsystem- bzw. BS2000/OSD-Versionen als unveränderliche Meldungs-Bestandteile garantiert, d.h. weder Meldungsschlüssel noch Nummern und Semantik der Inserts werden in künftigen Versionen verändert. Entsprechende Meldungen werden als **garantierte Meldungen** bezeichnet. Die Auszeichnung dieser Meldungen erfolgt über das Dienstprogramm MSGMAKER.
MIP erzeugt für garantierte Meldungen S-Variablen, über die direkt auf bestimmte Meldungsdaten zugegriffen werden kann; das Ausgabelayout der Meldungen muss nicht bekannt sein. Weitere Informationen hierzu siehe Abschnitt „Ausgabe von Meldungen in S-Variablen“ auf Seite 600.

Garantierte Meldungen werden durch das Meldungsattribut „Garantie“ (siehe Seite 592) dokumentiert.

*Beispiel*

Die Meldung `JMS0066`, eine garantierte Systemmeldung, wird nach erfolgreicher Bearbeitung des Kommandos ENTER-JOB ausgegeben. Sie enthält vier Inserts und hat folgendes Ausgabelayout:

```
%  JMS0066 JOB '(&00)' ACCEPTED ON (&01) AT (&02), TSN = (&03)
```

Über die Inserts wird der Auftragsname - (&00) -, das Datum - (&01) -, die Uhrzeit - (&02) - und die TSN - (&03) - des Jobs ausgegeben. Die Zuordnung der Insertnummern zu ihrem Inhalt ist garantiert (d.h. über die Insertnummer (&03) dieser Meldung wird immer eine TSN des Jobs (&00) ausgegeben werden). Darüber hinaus ist garantiert, dass bei einem erfolgreich abgesetzten Kommando ENTER-JOB immer die Meldung `JMS0066` ausgegeben wird. Nicht garantiert ist dagegen der Meldungstext, der in einer folgenden Systemversion evtl. auf folgende Art geändert werden kann:

```
%  JMS0066 TSN (&03): JOB '(&00)' ACCEPTED AT (&02) ON (&01)
```

Die Reihenfolge der Inserts hat sich zwar geändert, die logischen Zuordnungen der Inserts bleiben jedoch ebenso erhalten wie die Meldungsaussage der gesamten Meldung.

Folgende Meldungen von BS2000/OSD-BC sind garantierte Meldungen:

ACS0000
ACS0001
ACS0014
ACS0019
ACS0032
ACS0046
ACS0047
ACS0048
ACS0049
BLP0002
BLP0005
BLP0006
BLP0007
BLP0008
BLP0009
BLP0010
BLP0011
BLP0012
BLP0013
BLP0014
BLP0016
BLP0034
BLP0035
BLP0040
BLP0041
BLP0053
BLP0054
BLP0055
BLP0060
BLP0061
BLP0100
BLP0101
BLP0102
BLP0104
BLP0139
BLP0140
BLP0142
BLP0150
BLP0151
BLP0152
BLP0153
BLP0160
BLP0162
BLP0163
BLP0165
BLP0166
BLP0167
BLP0168
BLP0170
BLP0171
BLP0230
BLP0231
BLP0232
BLP0233
BLP0234
BLP0235
BLP0310
BLP0311
BLP0330
BLP0331
BLP0332
BLP0333
BLP0334
BLP0335
BLP0337
BLP0338
BLP0339
BLP0340
BLP0341
BLP0342
BLP0345
BLP0420
BLP0421
BLP0422
BLP0424
BLP0425
BLP0450
BLP0451
BLP0452
BLP0453
BLP0454
BLP0456
BLP0457
BLP0461
BLP0462
BLP0464
BLP0467
BLP0468
BLP0469
BLP0470
BLP0500
BLP0501
BLP0502
BLP0505
BLP0506
BLP0507
BLP0508
BLP0509
BLP0510
BLP0511
BLP0512
BLP0513
BLP0514
BLP0515
BLP0516
BLP0517
BLP0518
BLP0520
BLP0521
BLP0522
BLP0523
BLP0524
BLP0529
BLP0530
BLP0532
BLP0533
BLP0534
BLP0536
BLP0540
BLP0550
BLP0551
BLP0552
BLP0553
BLP0554
BLP0555
BLP0560
BLP0570
BLP0977
BLP0979
BLP0980
BLP0981
BLP0990
BLP0991
BLP0992
BLP0993
BLP0994
BLP0995
BLP0996
BLP0997
BLS0002
BLS0005
BLS0006
BLS0007
BLS0008
BLS0009
BLS0010
BLS0011
BLS0012
BLS0013
BLS0014
BLS0016
BLS0017
BLS0018
BLS0034
BLS0035
BLS0040
BLS0041
BLS0053
BLS0054
BLS0055
BLS0060
BLS0061
BLS0100
BLS0101
BLS0102
BLS0104
BLS0106
BLS0139
BLS0140
BLS0142
BLS0150
BLS0151
BLS0152
BLS0153
BLS0160
BLS0162
BLS0163
BLS0165
BLS0166
BLS0167
BLS0168
BLS0170
BLS0171
BLS0230
BLS0231
BLS0232
BLS0233
BLS0234
BLS0235
BLS0310
BLS0311
BLS0330
BLS0331
BLS0332
BLS0333
BLS0334
BLS0335
BLS0337
BLS0338
BLS0339
BLS0340
BLS0341
BLS0342
BLS0345
BLS0354
BLS0355
BLS0356
BLS0357
BLS0358
BLS0420
BLS0421
BLS0422
BLS0424
BLS0425
BLS0450
BLS0451
BLS0452
BLS0453
BLS0454
BLS0455
BLS0456
BLS0457
BLS0461
BLS0462
BLS0464
BLS0465
BLS0467
BLS0468
BLS0469
BLS0470
BLS0500
BLS0501
BLS0502
BLS0503
BLS0505
BLS0506
BLS0507
BLS0508
BLS0509
BLS0510
BLS0511
BLS0512
BLS0513
BLS0514
BLS0515
BLS0516
BLS0517
BLS0518
BLS0520
BLS0521
BLS0522
BLS0523
BLS0524
BLS0525
BLS0526
BLS0527
BLS0529
BLS0530

BLS0531
BLS0532
BLS0533
BLS0534
BLS0535
BLS0536
BLS0537
BLS0538
BLS0539
BLS0540
BLS0550
BLS0551
BLS0552
BLS0553
BLS0554
BLS0555
BLS0560
BLS0570
BLS0977
BLS0979
BLS0980
BLS0981
BLS0990
BLS0991
BLS0992
BLS0993
BLS0994
BLS0995
BLS0996
BLS0997
CJC0002
CJC0010
CJC0011
CJC0020
CJC0021
CJC0022
CJC0025
CJC0050
CJC0051
CJC0070
CJC0072
CJC0076
CMD0001
CMD0002
CMD0093
CMD0201
CMD0211
CMD0214
CMD0216
CMD0221
CMD0300
CMD0302
CMD0490
CMD0500
CMD0508
CMD0509
CMD0555
CMD0557
CMD0601
CMD0671
CMD0672
CMD0674
CMD0678
CMD0679
CMD0680
CMD0681
CMD0682
CMD0687
CMD0688
CMD0689
CMD0690
CMD2203
CMS0002
DMS0350
DMS0351
DMS0352
DMS0355
DMS035A
DMS035B
DMS035C
DMS035D
DMS0362
DMS0363
DMS0364
DMS036B
DMS036C
DMS036D
DMS036E
DMS037B
DMS0382
DMS03BE
DMS051A
DMS051B
DMS051C
DMS0533
DMS053C
DMS053F
DMS0574
DMS059C
DMS05BF
DMS05C3
DMS05C6
DMS05F8
DMS060A
DMS060B
DMS0666
DMS0681
DMS0682
DMS0683
DMS0684
DMS06D5
DMS0800
DMS0801
DMS0808
DMS0811
DMS0812
DMS0815
DMS0816
DMS0949
DMS09B5
DMS09B6
DMS0E49
DMS1343
DMS1373
DMS1374
DMS1376
DMS13C9
DMS13E2
DMS1D01
DMS1D02
DMS1D03
DMS1D04
DMS1D05
DMS1D06
DMS1D07
DMS1D08
DMS1D09
DMS1D0A
DMS1D0B
DMS1D0C
DMS1D0D
DMS1D0E
DMS1D0F
DMS1D10
DMS1D11
DMS1D12
DMS1D13
DMS1D1F
DSP0006
DSP0007
ECCCOPY
EMM2303
EMM2304
EMM2305
EMM2307
EMM2314
EMM2316
EMM2317
EMM2320
EMM2350
EMM2800
EMM2801
EMM2802
EMM2803
EMM2804
EMM2805
EMM2806
EMM2807
EMM2808
EMM2809
EMM2810
EMM2811
EMM2812
EMM2813
EMM2814
EMM2815
EMM2817
EMM2818
EMM2819
EMM2821
EMM2822
EMM2823
EMM2824
EMM2825
EMM2826
EMM2828
EMM2829
EMM2830
EMM2832
EMM2834
EMM2835
EMM2836
EMM2837
EMM2838
EMM2839
EMM2840
EMM2842
EMM2843
EMM2844
EMM2846
EMM2847
EMM2848
EMM2850
EMM2851
EMM2853
EMM2855
EMM2857
EMM2858
EMM2860
EMM2863
EMM2865
EMM2867
EMM2870
EMM2877
EMM2878
EMM2880
EMM2881
EMM2882
EMM2885
EMM2886
EMM2888
EMM2889
EMM2891
EMM2892
EMM2894
EMM2895
EMM2897
EMM2898
ESM0201
ESM0202
ESM0203
ESM0204
ESM0205
ESM0206
ESM0207
ESM0208
ESM0209
ESM0210
ESM0211
ESM0212
ESM0213
ESM0214
ESM0215
ESM0216
ESM0218
ESM0219
ESM0220
ESM0221
ESM0222
ESM0223
ESM0224
ESM0225
ESM0226
ESM0228
ESM0230
ESM0231
ESM0232
ESM0233
ESM0235
ESM0237
ESM0238
ESM0239
ESM0240
ESM0241
ESM0242
ESM0244
ESM0245
ESM0246
ESM0249
ESM0252

| | | | | | |
|---|---|---|---|---|---|
| ESM0253 | EXC0315 | IDA0300 | IMO9029 | JMS0024 | LMC0071 |
| ESM0254 | EXC0316 | IDA0301 | IMO9030 | JMS0025 | LMC0081 |
| ESM0256 | EXC0317 | IDH0101 | IMO9031 | JMS0043 | LMC0084 |
| ESM0257 | EXC0318 | IDH0102 | IMO9032 | JMS0044 | LMC0093 |
| ESM0259 | EXC031A | IDH0103 | IMO9033 | JMS0045 | LMC0095 |
| ESM0260 | EXC031F | IDH0104 | IMO9034 | JMS0046 | LMC0102 |
| ESM0262 | EXC032A | IDH0105 | IMO9035 | JMS0066 | LMC0129 |
| ESM0267 | EXC032B | IDH0106 | IMO9036 | JMS0067 | LMC0151 |
| ESM0269 | EXC0354 | IDH0107 | IMO9037 | JMS0100 | LMC0163 |
| ESM0288 | EXC0371 | IDH0108 | IMO9038 | JMS0101 | LMC0199 |
| ESM0292 | EXC0419 | IDH0109 | IMO9039 | JMS0105 | LMC0201 |
| ESM0322 | EXC0707 | IDH0110 | IMO9090 | JMS0110 | LMC0211 |
| ESM0332 | EXC0713 | IDH0111 | IMO9091 | JMS0111 | LMC0213 |
| ESM0432 | EXC0714 | IDH0112 | IMO9092 | JMS0113 | LMC0214 |
| ESM0600 | EXC0755 | IDH0113 | IMO9100 | JMS0130 | LMC0238 |
| ESM0603 | EXC0773 | IDH0114 | IMO9101 | JMS0131 | LMC0274 |
| ESM0604 | EXC0861 | IDH0115 | IMO9103 | JMS0133 | LMC0286 |
| ESM0610 | EXC0862 | IDH0116 | IMO9200 | JMS0143 | LMC0301 |
| ESM0612 | EXC0863 | IMO9001 | IMO9201 | JMS0171 | LMC0302 |
| ESM0613 | EXC0896 | IMO9002 | IMO9202 | JMS0264 | LMC0303 |
| ESM0614 | EXC0897 | IMO9003 | IMO9203 | JMS0265 | LMC0304 |
| ESM0615 | EXC0908 | IMO9004 | IMO9204 | JMS0268 | LMC0310 |
| ESM0670 | EXC0926 | IMO9005 | IMO9205 | JMS0522 | LMC0311 |
| ESM0671 | EXC0927 | IMO9006 | IMO9206 | JPM0204 | LMC0312 |
| ESM0672 | EXC0930 | IMO9007 | IMO9207 | JPM0205 | LMC0411 |
| ESM0673 | EXC0931 | IMO9008 | IMO9208 | JPM0500 | LMC0412 |
| ESM0674 | EXC0936 | IMO9009 | IMO9209 | JVS04A2 | LMC0413 |
| ESM0675 | GCF1014 | IMO9011 | IMO9210 | JVS04A3 | LMC0509 |
| EXC0012 | HEL0001 | IMO9012 | IMO9211 | JVS04D0 | LMC0510 |
| EXC0123 | HEL0002 | IMO9013 | IMO9212 | JVS04D1 | LMC0712 |
| EXC0155 | HEL0003 | IMO9014 | IMO9213 | JVS04D2 | LMC0714 |
| EXC0156 | HEL0004 | IMO9015 | IMO9214 | JVS04D3 | LMC0721 |
| EXC0269 | HEL0005 | IMO9016 | IMO9215 | JVS04D4 | LMC1002 |
| EXC0300 | HEL0006 | IMO9017 | JDS0301 | JVS04D5 | LMC1003 |
| EXC0303 | HEL0010 | IMO9018 | JDS0302 | JVS04D6 | LMC1004 |
| EXC0304 | HEL0011 | IMO9019 | JDS0303 | JVS04D8 | MCA0005 |
| EXC0305 | HEL0012 | IMO9020 | JDS0304 | JVS0465 | NAM0001 |
| EXC0307 | HEL0100 | IMO9021 | JDS0308 | JVS0468 | NAM0002 |
| EXC0309 | HEL0101 | IMO9022 | JDS0309 | JVS0469 | NAM0003 |
| EXC030B | HEL0102 | IMO9023 | JDS0310 | LMC0020 | NAM0004 |
| EXC030D | HEL0105 | IMO9024 | JDS0314 | LMC0035 | NAM0005 |
| EXC0310 | HEL0106 | IMO9025 | JDS0322 | LMC0036 | NAM0006 |
| EXC0311 | HEL0108 | IMO9026 | JMS0021 | LMC0041 | NAM0007 |
| EXC0312 | HEL0109 | IMO9027 | JMS0022 | LMC0053 | NAM0008 |
| EXC0313 | HEL0110 | IMO9028 | JMS0023 | LMC0064 | NAM0009 |

NAM0012
NAM0013
NAM0014
NAM0016
NAM0020
NAM0021
NAM0022
NAM0023
NAM0024
NAM0025
NAM0026
NAM0027
NAM0028
NAM0029
NAM0030
NAM0031
NAM0032
NAM0033
NAM0034
NAM0035
NAM0036
NAM0037
NAM0039
NAM0040
NAM0042
NAM0043
NAM0044
NAM0045
NAM0048
NAM3001
NAM3003
NBR0031
NBR0032
NBR0033
NBR0034
NBR0200
NBR0724
NBR0874
NBR0960
NBR0961
NBR0962
NBR1070
NBR1073
NBR1074
NDI0541
NER0000
NER1040
NER1060
NER1500
NER1510
NKA0094
NKA0096
NKD0002
NKD0005
NKD0006
NKD0007
NKD0009
NKD0010
NKD0011
NKD0013
NKA0094
NKA0096
NKD0002
NKD0005
NKD0006
NKD0007
NKD0009
NKD0010
NKD0011
NKD0013
NKD0014
NKD0016
NKD0018
NKD0019
NKD0021
NKD0023
NKD0024
NKD0025
NKD0026
NKD0027
NKD0028
NKD0030
NKD0031
NKD0032
NKD0033
NKD0034
NKD0035
NKD0036
NKD0037
NKD0038
NKD0039
NKD0040
NKD0041
NKR0018
NKR0073
NKR0074
NKR0075
NKR0124
NKR0125
NKR0147
NKR0148
NKR0149
NKR0150
NKR0151
NKR0160
NKR0161
NKR0162
NKR0163
NKR0164
NKR0165
NKR0166
NKR0167
NKR0168
NKR0169
NKR0170
NKR0171
NKR0172
NKR0173
NKR0174
NKR0175
NKR0176
NKR0177
NKR0178
NKR0179
NKR0180
NKR0181
NKR0182
NKR0188
NKR0189
NKR0191
NKR0192
NKR0193
NKR0194
NKR0195
NKR0196
NKR0197
NKR0201
NKR0202
NKS0008
NKS0010
NKS0011
NKS0012
NKS0014
NKS0015
NKS0036
NKS0046
NKS0047
NKS0053
NKS0055
NKS0056
NKS0057
NKS0062
NKS0063
NKS0068
NKS0069
NKS0071
NKV0001
NKV0004
NKV0005
NKV0006
NMH1102
NMH1103
NMH1104
NMH1105
NMH1106
NMH1108
NMH1109
NMH1116
NMH1117
NMH1120
NMH1123
NMH1124
NMH1154
NMH1155
NMH1165
NMH1166
NMH1180
NMH1181
NMH1183
PDT0200
PDT0201
PDT0202
PDT0203
PDT0204
PDT0205
PDT0206
PDT0207
PDT0208
PDT0209
PDT0210
PDT0213
PDT0214
SDP0090
SDP0091
SDP0092
SDP0099
SDP0140
SDP0517
SDP1008
SDP1018
SDP1030
SPF0300
SPF0680
SPF0681
SPF0682
SPF0688
SPF0689
SRM0003
SRM0004
SRM0016
SRM2101
SRM2103
SRM2151
SRM2210
SRM2211
SRM2401
SRM3102
SRM3999
SRM6001
SRM6010
SRM6020
SRM6030
SRM6040
SSM2012
SSM2039
SSM2052
SSM2053
SSM3056
SSM3100
SSM3101
SSM3284

## 15.4 Meldungssuche

Die Meldungssuche erfolgt zuerst im task-eigenen Speicherbereich; die Suchkriterien sind der Meldungsschlüssel und das im Makro angegebene Sprachkennzeichen. Bleibt im Taskbereich die Suche ohne Erfolg, wird auf Systemebene weitergesucht. Existiert auch dort keine Meldung zu dem angegebenen Meldungsschlüssel und der Sprache, wird die Meldungssuche zuerst mit der taskweit eingestellten Sprache, dann mit der ersten und schließlich mit der zweiten systemweit eingestellten Sprache fortgesetzt. Bleibt die Suche ohne Erfolg, wird die Meldung „MESSAGE UNDEFINED“ ausgegeben.
Bild 31 auf Seite 587 skizziert diesen Vorgang, ohne alle in der Beschreibung genannten Einzelheiten abzubilden.

#### Meldungssuche auf Taskebene

Auf Taskebene wird als Erstes der task-eigene Meldungspuffer durchsucht, in welchem die zuletzt ausgegebenen Meldungen abgespeichert sind. Ist die angeforderte Meldung dort abgelegt, wird sie aufbereitet und ausgegeben.
Andernfalls wird überprüft, ob die Bereichszuordnungsliste (Class List) die angegebene Meldungsklasse enthält. Die Class List enthält zu allen „taskbekannten“ Meldungsklassen die Namen der entsprechenden Meldungsdateien. Besonders häufig auszugebende Meldungen werden in der Meldungsdatei mit DLAM-Zugriff spezifiert. Sie werden zum Zeitpunkt der Verknüpfung der Meldungsdatei mit der Task in den Task-DLAM-Bereich übertragen. Ist die Meldungsklasse vorhanden, so wird im nächsten Schritt die Meldungsdatei nach dem Meldungsschlüssel und dem Sprachkennzeichen durchsucht.
Wird die Meldung gefunden, so wird der taskeigene Meldungspuffer mit dem „Meldungsgerüst“ aktualisiert, das „Meldungsgerüst“ mit den vom Makro MSG7/MSG7X übergebenen aktuellen Einfügungen ergänzt und die Meldung ausgegeben.

#### Meldungssuche auf Systemebene

Ist die Meldung in keiner task-eigenen Meldungsdatei zu finden, wird der Suchvorgang auf Systemebene im systemeigenen Meldungspuffer fortgesetzt. Entspricht auch hier keine Meldung den Vorgaben, wird die MIP-Task aufgerufen, um mit der Meldungssuche in den Systemmeldungsdateien fortzufahren. Auch hier wird, analog zur Task-Ebene, zuerst der DLAM-Bereich überprüft, bevor die Meldungsdateien durchsucht werden.

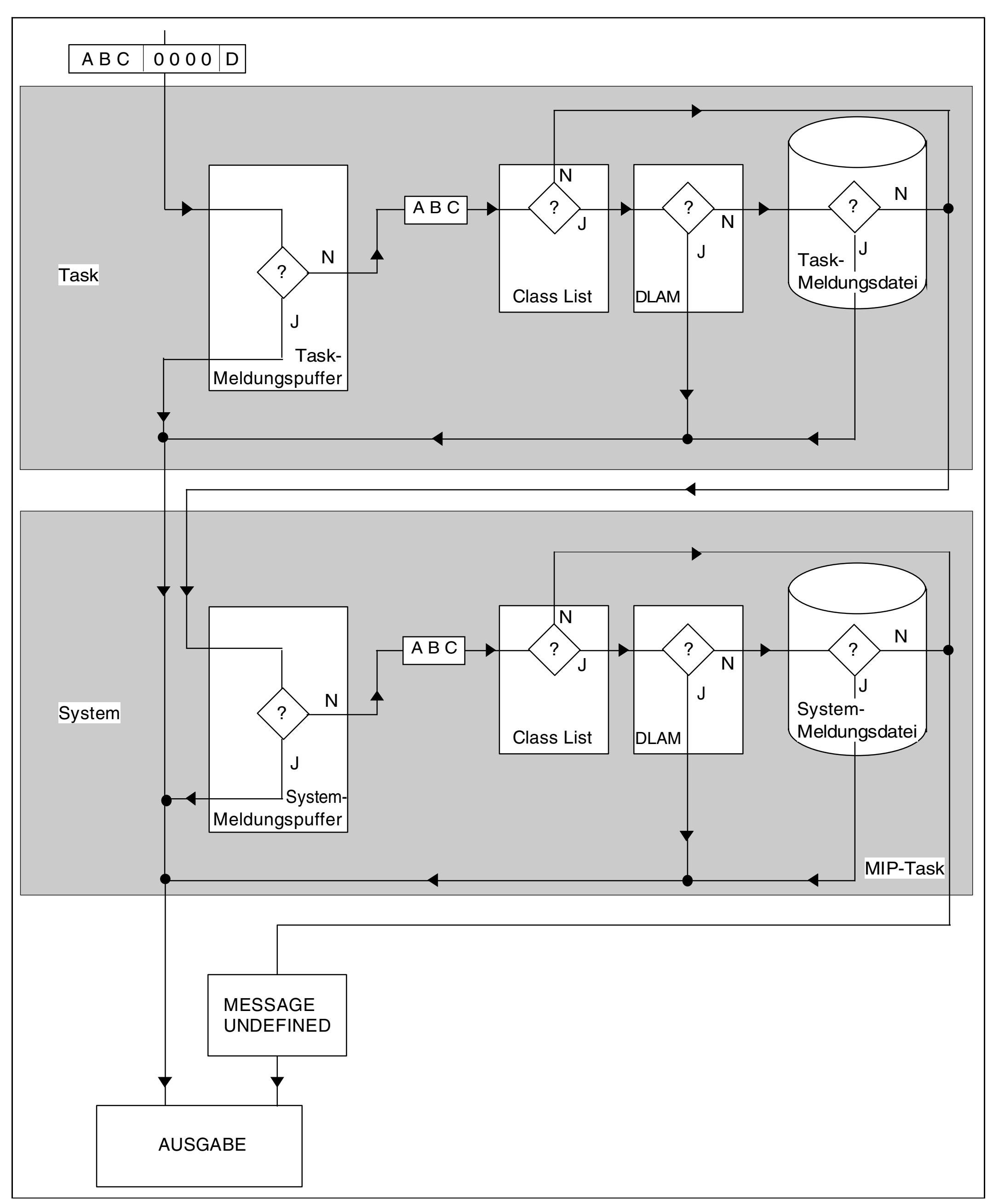
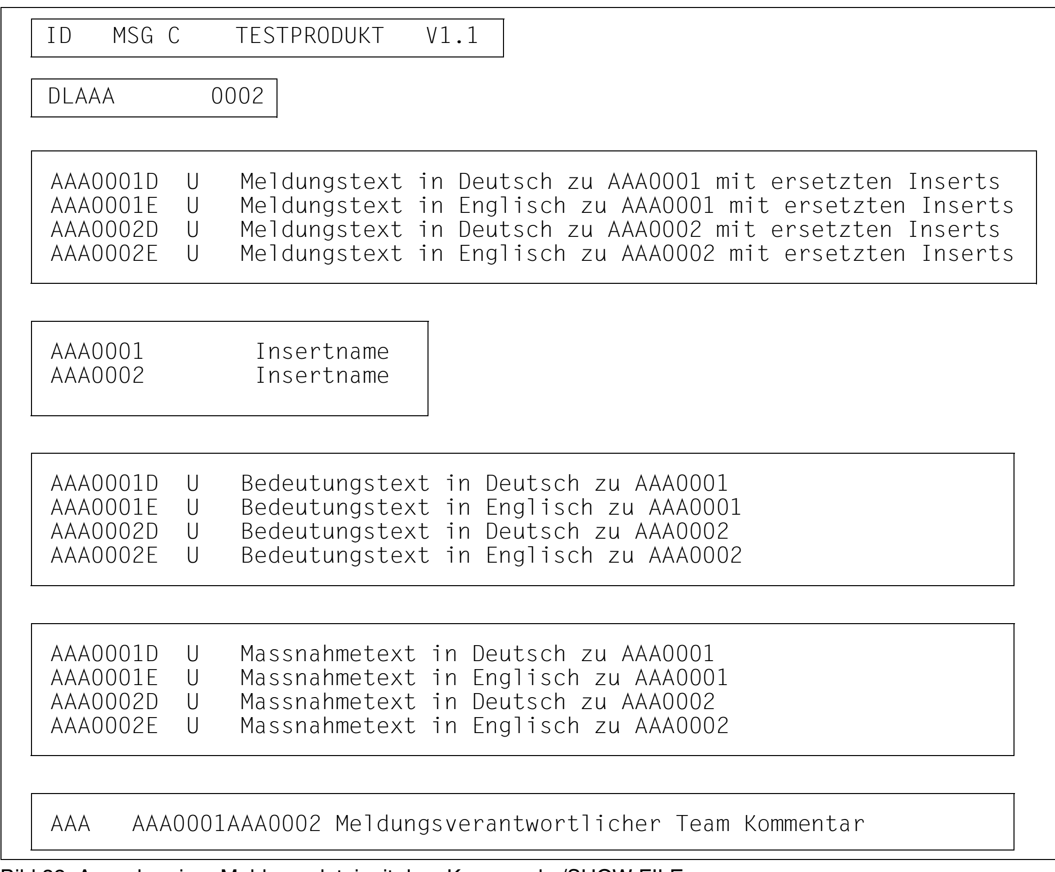



Bild 31: Meldungssuche im BS2000

# 15.5 Struktur einer Meldungseinheit

Jede mit dem Dienstprogramm MSGMAKER erstellte Meldungsdatei setzt sich aus einzelnen Meldungseinheiten zusammen, auf die über einen siebenstelligen Meldungsschlüssel zugegriffen wird. Die Meldungseinheit besteht aus Meldungsattributen, Meldungstext und ggf., um ein in der Meldung zitiertes Problem näher zu erläutern und mögliche Maßnahmen aufzuzeigen, einem Bedeutungs- und Maßnahmetext. Alle Texte können in maximal acht Sprachen eingetragen werden. Jede Sprache wird durch ein einstelliges Buchstabenkürzel, dem Kennbuchstaben für die Sprache, dargestellt.
Über die zugeordneten Meldungsattribute kann die Meldungsausgabe gesteuert werden.

Es folgt ist der Aufbau einer Meldungsdatei, wie sie mit dem Kommando `/SHOW-FILE <meldungsdatei>` zu sehen ist. Nach der Identifikationszeile erscheinen Meldungen, die für den DLAM-Zugriff spezifiziert sind. Sie werden zum Zeitpunkt der Verknüpfung der Meldungsdatei mit dem System/der Task in den System/Task-DLAM-Bereich übertragen.



```
ID   MSG C    TESTPRODUKT   V1.1

DLAAA       0002

AAA0001D  U   Meldungstext in Deutsch zu AAA0001 mit ersetzten Inserts
AAA0001E  U   Meldungstext in Englisch zu AAA0001 mit ersetzten Inserts
AAA0002D  U   Meldungstext in Deutsch zu AAA0002 mit ersetzten Inserts
AAA0002E  U   Meldungstext in Englisch zu AAA0002 mit ersetzten Inserts

AAA0001        Insertname
AAA0002        Insertname

AAA0001D  U   Bedeutungstext in Deutsch zu AAA0001
AAA0001E  U   Bedeutungstext in Englisch zu AAA0001
AAA0002D  U   Bedeutungstext in Deutsch zu AAA0002
AAA0002E  U   Bedeutungstext in Englisch zu AAA0002

AAA0001D  U   Massnahmetext in Deutsch zu AAA0001
AAA0001E  U   Massnahmetext in Englisch zu AAA0001
AAA0002D  U   Massnahmetext in Deutsch zu AAA0002
AAA0002E  U   Massnahmetext in Englisch zu AAA0002

AAA   AAA0001AAA0002 Meldungsverantwortlicher Team Kommentar
```

Bild 32: Ausgabe einer Meldungsdatei mit dem Kommando /SHOW-FILE

# 15.6 Meldungsschlüssel

Der Meldungsschlüssel ist siebenstellig, wobei die ersten drei Buchstaben für die Meldungsklasse und die letzten vier Zeichen für die Meldungsnummer stehen.
Die Meldungsklasse bezeichnet eine BS2000-Komponente, ein Subsystem, Modul o. Ä.; die Meldungsnummer dient der laufenden Nummerierung innerhalb einer Meldungklasse. In der Regel besteht die Meldungsnummer aus vier Ziffern. Meldungen, deren Meldungsnummern auch Buchstaben enthalten, werden in die Meldungsdatei vor solchen Meldungen eingefügt, welche Ziffern beinhalten.

*Beispiel*

Die Meldungsschlüssel ABC123A bis ABC123Z werden vor den Meldungsschlüsseln ABC1230 bis ABC1239 in die Meldungsdatei eingefügt.

## 15.6.1 Kennbuchstabe für die Sprache

Ein alphabetisches Zeichen dient zur Kennzeichnung der Sprache, in welcher Meldungs-, Bedeutungs- und Maßnahmetext erstellt sind. Der Kennbuchstabe D wird für Deutsch und E für Englisch verwendet, alle weiteren Sprachen sind mit beliebigen Buchstaben abkürzbar. Maximal acht Sprachen können definiert werden. Der Kennbuchstabe dient in Verbindung mit dem Meldungsschlüssel zur Identifikation eines Meldungs-, Bedeutungs- und Maßnahmetextes innerhalb der Meldungseinheit.

## 15.6.2 Attribute der Meldungseinheit

### MIP-Zugriffsmethode für die Meldung

Es gibt folgende MIP-Zugriffsmethoden für Meldungen: ISAM, DLAM, LOCAL-DLAM, BAMR und MINIMIP. Die Zugriffsmethoden LOCAL-DLAM, BAMR und MINIMIP sind für den internen Gebrauch reserviert.

### Ausgabeziel der Meldung

Dokumentiert ein oder mehrere Ausgabeziele der Meldung. Die Angabe wird nicht ausgewertet, sollte jedoch mit dem Operanden DEST des Makros MSG7/MSG7X übereinstimmen.

### Berechtigungsschlüssel (Routing-Code)

Ist das Ausgabeziel einer Meldung eine Bedienstation, erhält sie ein zusätzliches, einstelliges Kennzeichen, den so genannten Berechtigungsschlüssel (BS), auch Routing-Code (RC) genannt. Er wird bei den Meldungen als Zielangabe ausgewertet, wenn neben der Hauptbedienstation auch Nebenbedienstationen verwendet werden.

Beim Betrieb mit mehreren Bedienstationen werden diese bestimmten Aufgabengebieten zugeteilt. Über den Berechtigungsschlüssel können dann bestimmte Meldungen vom System den richtigen Bedienstationen zugeordnet werden.
Der mit MSGMAKER definierte Berechtigungsschlüssel wird nur verwendet, wenn dem Operanden UCDEST im Makro MSG7/MSG7X kein Wert zugewiesen wird.

| BS / RC | Aufgabengebiet |
|---|---|
| A | Systemverwaltung |
| C | Bedienung des Datenkommunikationssystems |
| D | Bedienung von Plattenspeichern |
| E | Allgemeine Aufgaben und Befugnisse |
| G | Geräteverwaltung |
| H | Hardware-Wartung |
| J | Steuerung und Kontrolle von Jobs |
| K | OPR-Kommandoverwaltung |
| N | Überwachung von Remote-Spool |
| O | Bedienung von Schnelldruckern |
| P | Steuerung von Tasks |
| R | Überwachung und Steuerung des Systemverhaltens |
| S | Steuerung des SPOOLOUT-Betriebs, SPOOL laden |
| T | Bedienung von Magnetbandgeräten |
| U | Dateiverwaltung |
| W,X,Y,Z | Zur freien Verfügung für den Benutzer |
| 0 | Meldungen mit diesem Berechtigungsschlüssel sind nicht bestellbar. Meldungen ohne Antwort werden nur in der CONSLOG-Datei hinterlegt. Meldungen mit Antwort (Fragen) werden der Hauptbedienstation zugestellt. |
| 9 | VM2000-Verwaltung |
| # | POSIX |
| * | Hauptbedienstation |
| @ | Hauptbedienstation (nur bei Meldungen, die eine Antwort erfordern; Meldungen ohne Antwort werden in der CONSLOG-Datei hinterlegt, aber nicht ausgegeben) |
| $ | Sonderbedeutung; wird nicht für Meldungsausgaben verwendet |
| B,F,I,L,M,Q,V,<br>1,2,3,4,5,6,8 | Reserviert für künftige Systemerweiterungen |

Tabelle 46: Aufgabengebiete und Berechtigungsschlüssel (BS / RC) von Meldungen

Siehe auch Tabelle 49 auf Seite 642.

### Weight Code

Der Weight Code bestimmt das Meldungsgewicht.

Ist das Ausgabeziel einer Meldung eine Bedienstation, muss ein Weight Code vergeben werden, andernfalls ist eine Zuweisung ohne Bedeutung. Meldungen, die nicht an eine Bedienstation gehen, kann WEIGHT=*NONE zugewiesen werden. Wird eine Meldung mit anders lautender Zielangabe trotzdem an eine Bedienstation geschickt, erhält sie vom Makro $TYPIO den Standard Weight Code 99.

Werte von 0 bis 99 können vergeben werden, wobei Weight Code 99 einer Meldung äußerste Wichtigkeit bescheinigt. Die Meldung kann nicht unterdrückt werden.
Erfordert die Meldung eine Antwort (Operand REPLY im Makro MSG7/MSG7X) sollte der Weight Code 90 oder 98 zugewiesen werden.

Zur Filterung von Meldungen können an der Bedienstation per Kommando vier Filterstufen eingestellt werden, welche den Weight Codes der Meldungen folgendermaßen zugeordnet sind:

| Filterstufe | Weight Code | Standardwerte des Herstellers |
|---|---|---|
| 1 | 0 - 19 | Keine |
| 2 | 20 - 39 | 30 |
| 3 | 40 - 59 | 50 |
| 4 | 60 - 79 | 70 |
| nicht unterdrückbar | 80 - 99 | 90,98,99 |



Die in den ausgelieferten Meldungsdateien eingestellten Meldungsgewichte (Vorbesetzung) können von der Systembetreuung über die MSGMAKER-Anweisung //MODIFY-MSG geändert werden.

*Filterstufe 1*

Der Benutzer kann nachträglich Meldungen, die seiner Meinung nach bedeutungslos sind, mit einem Gewicht dieses Bereichs versehen und durch Zuordnung der Filterstufe 1 für die betreffende Bedienstation/den berechtigten Benutzer-Job eine dortige Ausgabe verhindern.

*Filterstufe 2*

Eine Meldung mit einem Weight Code aus diesem Bereich ist rein informativ und kann ohne Gefährdung des sicheren Betriebs unterdrückt werden.

*Filterstufe 3*

Eine Meldung aus diesem Bereich ist informativ. Der sichere Betrieb wird nicht gefährdet, wenn sie unterdrückt wird. Der erhöhte Wert erklärt sich durch einen Zusatzaspekt, durch den eine Ausgabe ratsam sein kann, z.B. Protokollierung aus Datenschutzgründen, Bearbeitung durch einen berechtigten Benutzer-Job.

*Filterstufe 4*

Eine Meldung mit einem Weight Code aus diesem Bereich ist informativ und erfordert keine Antwort. Aber die Information ist für den Operator wichtig. Ihre Unterdrückung könnte unter Umständen negative Auswirkungen auf den Systemlauf haben.

*Weight Code-Bereich 80 - 99*

Eine Meldung mit einem Weight Code dieses Bereichs erfordert entweder eine Antwort oder ist von größter Wichtigkeit.

- Weight Code 90
  Mit diesem Gewicht werden Antwortmeldungen versehen, für die eventuell später in Verbindung mit gesicherten Automatisierungsfunktionen eine Unterdrückung denkbar wäre. Zum aktuellen Zeitpunkt sind sie nicht unterdrückbar.
- Weight Code 98
  Eine Meldung mit diesem Weight Code erfordert eine Antwort und darf in keinem Fall unterdrückt werden. Weight Code 98 ist von der Wichtigkeit her gleichzusetzen mit dem Weight Code 99.
- Weight Code 99
  Eine Meldung mit diesem Weight Code ist von höchster Wichtigkeit und darf in keinem Fall unterdrückt werden.

### Garantie

Das Meldungsattribut „Garantie“ wird von MIP ausgewertet.

Das Attribut garantiert, dass bei einer damit versehenen Meldung weder Meldungsschlüssel noch Inserts (Nummern und Semantik) in den kommenden Subsystem- bzw. BS2000/OSD-Versionen geändert werden.

MIP erzeugt für garantierte Meldungen S-Variablen. Nähere Informationen hierzu siehe Abschnitt „Ausgabe von Meldungen in S-Variablen“ auf Seite 600.

### 15.6.3 Meldungstext

Zu jedem definierten Meldungsschlüssel muss ein Meldungstext erstellt werden, der in acht Sprachen abgefasst werden kann. In den Meldungstext können maximal 30 Inserts (siehe unten) eingefügt werden.
Der Meldungstext ist mit dem Meldungsschlüssel und Kennbuchstaben für die Sprache in der Meldungsdatei abgespeichert (siehe Bild 32 auf Seite 588).
Bei der Erstellung eines Meldungstextes sind folgende Besonderheiten zu beachten:

- Trennungszeichen „^“

  Der Meldungstext wird als fortlaufende Textzeile eingegeben. Um eine strukturierte Ausgabe zu erhalten, kann das Trennungszeichen „^“ an jeder beliebigen Stelle und beliebig oft in den Meldungstext eingefügt werden. Text, der auf dieses Trennungszeichen folgt, wird in der jeweils nächsten Zeile ausgegeben. Mit der SHOW-Funktion von MSGMAKER kann die Textausgabe überprüft werden.

- Beschränkung des Zeichenvorrats

  Die Meldungsausgabe kann auf unterschiedlichen 7-Bit-Terminals erfolgen. Abweichende Codier-Tabellen haben eine doppelte Belegung von Zeichen zur Folge.



Folgende Zeichen sind sowohl im Meldungstext als auch im Bedeutungs- und Maßnahmetext verboten, wenn die Texte in den Standardsprachen Deutsch und Englisch erstellt sind. MSGMAKER gibt eine Warnung aus, falls eines dieser Zeichen verwendet wurde. Die Einschränkungen gelten nicht für alle zusätzlich definierten Sprachen.

| **Code** | **Zeichen** | | | | | | |
|---|---|---|---|---|---|---|---|
| Hex-Code X' __' | 4F | BB | BC | BD | FB | FD | FF |
| Internationaler<br>EBCDIC.DF.03 | l | \ | [ | ] | < | > | ~ |
| Deutscher<br>EBCDIC.DF.03 | ö | Ö | Ä | Ü | ä | ü | ß |

Leerzeichen am Ende des Meldungstextes werden unterdrückt. Der letzte Satz oder Satzteil eines Meldungstextes schließt ohne Satzzeichen ab.

- Inserts (Einfügungen)

*Definition von Inserts*

Maximal 30 verschiedene Inserts können sprachübergreifend in den Meldungstexten einer Meldungseinheit definiert werden.
Die Zeichenfolgen (&00), (&01) bis (&29) werden in den Meldungstext eingefügt, wobei 00 bis 29 die Nummern der Inserts darstellen. Sie können im Text in beliebiger Reihenfolge auftreten (00, dann 02, dann 01) und müssen nicht lückenlos definiert werden (00, 08, 11).
Entscheidend für die Meldungsausgabe ist, dass die definierten Inserts im Makro MSG7/MSG7X aufgeführt werden. Enthält der Meldungstext mehr Inserts als im Makroaufruf angegeben, wird für jede übrige Einfügung der Standard-Text übernommen. Ist kein Standard-Text vorhanden, unterscheidet man zwei Fälle:

1. Die Nummer des übrigen Inserts (z.B. (&02)) liegt zwischen den Insertnummern, die im Makroaufruf angegeben sind (z.B. (&00),(&01),(&03)). In diesem Fall wird ein Leerstring ausgegeben.
2. Die Nummer des übrigen Inserts (z.B. (&03)) ist höher als die Insertnummern, die im Makroaufruf angegeben sind (z.B. (&00),(&01),(&02)). Der Insert wird nicht ersetzt, d.h. es wird (&03) ausgegeben.

Ist im Makro MSG7/MSG7X eine Angabe zu einer Insertnummer gemacht, die im Meldungstext nicht definiert ist, ignoriert MIP die Angabe im Makro MSG7/MSG7X. Es erscheint keine Fehlermeldung.
Die mit MSGMAKER in der Meldung angegebenen Standard-Texte ersetzen die definierten Inserts nur, wenn im Makro MSG7/MSG7X kein aktueller Text dafür vereinbart ist.

*Insertnamen*

Für jeden Insert kann zusätzlich ein mnemotechnischer Insertname vergeben werden. Insertnamen werden zur Bildung von S-Variablen verwendet.
Die vergebenen Insertnamen werden immer in Großbuchstaben umgewandelt. Weitere Informationen zur Bildung von S-Variablen siehe Abschnitt „Ausgabe von Meldungen in S-Variablen“ auf Seite 600.

*Standard-Text für Insert*

Text, der an Stelle der Inserts (&nn) in den Ausgabetext eingesetzt wird. Der Standard-Text ist Inhalt der S-Variablen I0 (bis I29).

*Automatic help*

Die automatische Hilfefunktion (Funktion /HELP-MSG-INFORMATION) wird für den Teil des Meldungsschlüssels aktiviert, der als Insertwert geliefert wird.



*Darstellung von Inserts in Anweisungen*
Wird ein Insert mit einer Anweisung (//ADD-MSG bzw. //MODIFY-MSG) in einem Meldungstext vereinbart, muss das Zeichen & doppelt eingegeben werden.

- Länge des Meldungstextes

  Der Meldungstext kann maximal 220 Zeichen, einschließlich der Zeichenfolgen (&00) ... (&29), lang sein. Die aktuellen Inhalte, welche bei der Meldungsausgabe diese Zeichenfolgen ersetzen sollen, sind in dieser Längenangabe nicht inbegriffen. Der vollständige Meldungstext kann folglich mehr als 220 Zeichen umfassen.

*Beispiel für einen Meldungstext*

```
MSG-TEXT =THIS MESSAGE WILL BE SENT TO THE DESTINATION "(&04)",^
IN LANGUAGE "(&00)", "(&01)" OR "(&02)"
```

*Maximale Textlänge bei der Meldungsausgabe auf die Datensichtstation*

Die maximale Textlänge, die von MSG7/MSG7X ausgegeben wird, darf 4079 Zeichen nicht übersteigen. Dies beeinhaltet den Meldungstext (ohne Meldungsschlüssel), die Insertwerte (ggf. die Standardwerte), die Datums- und Zeitangabe (wenn gewünscht).

Generell gilt: Übersteigt ein Meldungstext die maximale Textlänge, so wird dieser Text bei der Ausgabe abgeschnitten. Dabei gibt MSGMAKER keine Warnung aus.

*Maximale Textlänge bei der Meldungsausgabe auf die Bedienstation*

Die maximale Textlänge, die von MSG7/MSG7X an die Bedienstation ausgegeben wird, beträgt 230 Zeichen, einschließlich des vorangestellten Prozentzeichens % und des Meldungsschlüssels.

MSGMAKER überprüft die Länge des Meldungstextes und der Standard-Inserts. Bei Überschreitung der Textlänge von 230 Zeichen gibt MSGMAKER eine Warnung aus.

Bedingt durch die aktuellen Insertwerte oder die Optionen des Makros MSG7/MSG7X kann die Textlänge variieren. Bei Angabe von TEXT-ONLY wird der Meldungsschlüssel nicht ausgegeben, durch Angabe von DATESTAMP und TIMESTAMP kann die Datums- und Zeitangabe vor dem Meldungstext variieren.

### 15.6.4 Bedeutungs-/Maßnahmetext

Zur Erläuterung des Meldungstextes kann ein erklärender Text formuliert werden, der auf Anforderung ausgegeben wird. Weiterhin können Antworten oder Maßnahmen angeboten werden, mit deren Hilfe das im Meldungstext beschriebene Problem gelöst werden kann. Bedeutungs- und Maßnahmetexte werden in der vorher festgelegten Sprache ausgegeben.

Über das Kommando /HELP-MSG-INFORMATION wird ein solcher Bedeutungs- und Maßnahmetext nach SYSOUT ausgegeben (siehe Seite 598).
Sowohl Bedeutungs- als auch Maßnahmetext können aus maximal 256 Zeilen zu je 74 Zeichen bestehen. Inserts sowie das Trennungszeichen „^“ werden im Text akzeptiert, haben jedoch nicht die Sonderbedeutung, die sie im Meldungstext besitzen.
Die letzte Zeile des Bedeutungs- und Maßnahmetextes sollte mit einem Punkt enden.

Analog zum Meldungstext gelten die Beschränkungen des Zeichenvorrats.

## 15.7 Meldungsausgabe

### 15.7.1 Ausgabeformen der Meldungen

Als Ausgabeform für Meldungen kann generell gewählt werden:

- **Normalform**
  umfasst den Meldungsschlüssel, Meldungstext ggf. mit aktuellen Einfügungen
- **Kurzform**
  umfasst den Meldungsschlüssel und ggf. die aktuellen Einfügungen, aber keinen weiteren Meldungstext
- **Langform**
  umfasst die Normalform plus Erläuterungstext (Bedeutungs- und Maßnahmetext)

Die Ausgabeform der Meldungen kann der Anwender über den Operanden INFORMATION-LEVEL des Kommandos /MODIFY-JOB-OPTIONS ändern. Vereinbarungen, die mit /SET-LOGON-PARAMETERS oder /ENTER-JOB getroffen wurden und die die Protokollierung eines Jobs betreffen, werden überschrieben.

INFORMATION-LEVEL=*MEDIUM bewirkt die ungekürzte Ausgabe der Meldungen (Normalform).

INFORMATION-LEVEL=*MINIMUM bewirkt die Ausgabe der Meldungen in codierter Form (Kurzform).

**Beispiel**

Zur Demonstration der unterschiedlichen Ausgabeformen von Systemmeldungen wird im Job JOB.TEST versucht, die Attribute der nicht katalogisierten Datei ABCD auszugeben. Die Systemmeldungen werden in Normal- und Kurzform in die Datei SYSOUT.JOB.TEST protokolliert.
Der Job wird mit dem Kommando /ENTER-JOB FROM-FILE=JOB.TEST gestartet.

*Job JOB.TEST*

```
/SET-LOGON-PARAMETERS
/ASSIGN-SYSOUT TO=SYSOUT.JOB.TEST
/SHOW-FILE-ATTRIBUTES FILE-NAME=ABCD
/SET-JOB-STEP
/MODIFY-JOB-OPTIONS INFORMATION-LEVEL=*MINIMUM
/SHOW-FILE-ATTRIBUTES FILE-NAME=ABCD
/EXIT-JOB
```

*SYSOUT-Protokoll SYSOUT.JOB.TEST*

```
/SHOW-FILE-ATTRIBUTES FILE-NAME=ABCD

%  DMS0533 REQUESTED FILE NOT CATALOGED IN PUBSET '2OSG'. COMMAND TERMINATED
%  CMD0205 ERROR IN PRECEDING COMMAND OR PROGRAM AND PROCEDURE STEP
TERMINATION ------------------------------------------------------------  (1)

/SET-JOB-STEP

/MODIFY-JOB-OPTIONS INFORMATION-LEVEL=*MINIMUM -------------------------  (2)

/SHOW-FILE-ATTRIBUTES FILE-NAME=ABCD

%  DMS0533 2OSG
%  CMD0205 -------------------------------------------------------------  (3)

/EXIT-JOB
%  EXC0419 /LOGOFF AT <time> ON <date> FOR TSN '2EZD'
%  EXC0421 CPU TIME USED: 0.5752
```

(1) Ausgabe der Systemmeldungen in Normalform, da INFORMATION-LEVEL= *MEDIUM Voreinstellung ist.

(2) Ausgabeform wird auf INFORMATION-LEVEL=*MINIMUM gesetzt.

(3) Ausgabe der Systemmeldungen in codierter Kurzform.

## 15.7.2 Kommando HELP-MSG-INFORMATION

Dieses Kommando wird verwendet, um sich zu einer erhaltenen Meldung Erläuterungstexte (Bedeutungs- und Maßnahmetext) ausgeben zu lassen. Das Kommando kann ohne und mit Operanden abgesetzt werden (siehe Handbuch „Kommandos“ [27]).

**/HELP-MSG-INFORMATION ohne Operanden** gibt die vom System zuletzt ausgegebene Meldung (ggf. mit aktuellen Einfügungen) zusammen mit dem Bedeutungs- und Maßnahmetext in der gleichen Sprache nochmals aus.

**/HELP-MSG-INFORMATION mit Operanden** ermöglicht die gezielte Angabe eines Meldungsschlüssels (Operand MSG-IDENTIFICATION), des Informationsumfangs (Operand INFORMATION-LEVEL) und der Sprache (Operand LANGUAGE).

*Beispiel*

```
/help-msg-information msg-id=dms0533
%  DMS0533 REQUESTED FILE NOT CATALOGED IN PUBSET '(&00)'. COMMAND TERMINATED
%  ? This message is issued by DMS commands. The requested file is not
%    cataloged in the requested pubset.
%  RESPONSE : NONE % ------------------------------------------------  (1)

/help-msg-information msg-id=dms0533,language=d
%  DMS0533 ANGEGEBENE DATEI IN PUBSET '(&00)' NICHT GEFUNDEN. KOMMANDO
%  BEENDET
%  ? Diese Meldung wird von DVS-Kommandos ausgegeben. Die angeforderte Datei
%    ist im gewuenschten Pubset nicht katalogisiert.
%  MASSNAHME : KEINE ------------------------------------------------  (2)

/help-msg-information msg-id=dms0533,inf-level=*medium
%  DMS0533 REQUESTED FILE NOT CATALOGED IN PUBSET '(&00)'.
%  COMMAND TERMINATED -----------------------------------------------  (3)

/help-msg-information msg-id=dms0533,inf-level=*minimum
%  DMS0533 (&00) ----------------------------------------------------  (4)
```

(1) Ausgabe der Systemmeldung in Langform. INFORMATION-LEVEL=*MAXIMUM ist Voreinstellung.

(2) Englisch wurde als Standardwert für die Sprache festgelegt. Wird die Ausgabe der Meldung in Deutsch gewünscht, muss dies extra angegeben werden.
Mit dem Kommando /MODIFY-MSG-ATTRIBUTES kann der Anwender für seine Task die Sprache vereinbaren, in der die Meldungen ausgegeben werden sollen.

(3) Ausgabe der Systemmeldung in Normalform, Sprache Englisch ist Standardwert.

(4) Ausgabe der Systemmeldung in codierter Kurzform.

### 15.7.3 Meldungen, die eine Antwort erfordern

Bei Meldungen, die eine Stellungnahme erfordern, bietet MIP als Standard-Antwort ein Fragezeichen an. Das Fragezeichen ist ein reserviertes Schlüsselwort bei MIP. Setzt der Benutzer ein Fragezeichen ab, gibt MIP den Bedeutungs- und Maßnahmetext zu dieser Meldung aus und wiederholt die Frage.

```
%  CMD0213 TERMINATE PROGRAM? (Y=YES; N=NO)
? ————————————————————————————————————————————————————————————————————  (1)

%  ? The user requested program termination by hitting a
%    function key or by NEXT line input.
%  ! Y: causes the END statement to be executed
%    N: termination request is ignored
%  CMD0213 TERMINATE PROGRAM? (Y=YES; N=NO) ———————————————————————————  (2)
```

(1) Der Benutzer gibt ein Fragezeichen ein. MIP gibt den Bedeutungs- und Maßnahmetext zu dieser Meldung aus.

(2) Der im Anschluss daran erneut ausgegebene Meldungstext lässt sich vom Anwender nun leichter beantworten.

### 15.7.4 Ausgabe von Meldungen an Benutzerprogramme

Benutzern ist es möglich, sich mit dem Makroaufruf MSG7/MSG7X Meldungen vom System in einen Speicherbereich ihres Benutzerprogramms schreiben zu lassen (Operand BUFFER). Nähere Informationen siehe Handbuch „Makroaufrufe an den Ablaufteil“ [30].

### 15.7.5 Ausgabe von Meldungen in S-Variablen

Meldungen können in die Systemdateien SYSOUT bzw. SYSLST ausgegeben werden. Das kostenpflichtige Produkt SDF-P bietet ergänzend hierzu die Möglichkeit, **garantierte Meldungen** in strukturierte S-Variablen auszugeben (siehe auch Abschnitt „Garantierte Meldungen“ auf Seite 581).
SDF-P übernimmt dabei die Steuerung des S-Variablenstroms SYSMSG, über den die garantierten Meldungen in S-Variablen umgelenkt werden. Nähere Informationen zum Konzept der strukturierten S-Variablenströmen siehe Handbuch „SDF-P“ [47].
Die Systemkomponente MIP erzeugt die S-Variablen, deren Werte als Eingabedaten in S-Prozeduren weiterverwendet werden können.

Das Bild unten zeigt sowohl die Ausgabe nach SYSOUT durch den Systemdateien-Manager als auch die Umlenkung der Ausgabeinformation in die S-Variablenströme SYSMSG (und SYSINF), welche durch SDF-P gesteuert wird. Weitere Informationen zu SYSINF siehe Handbuch „Kommandos“ [27].

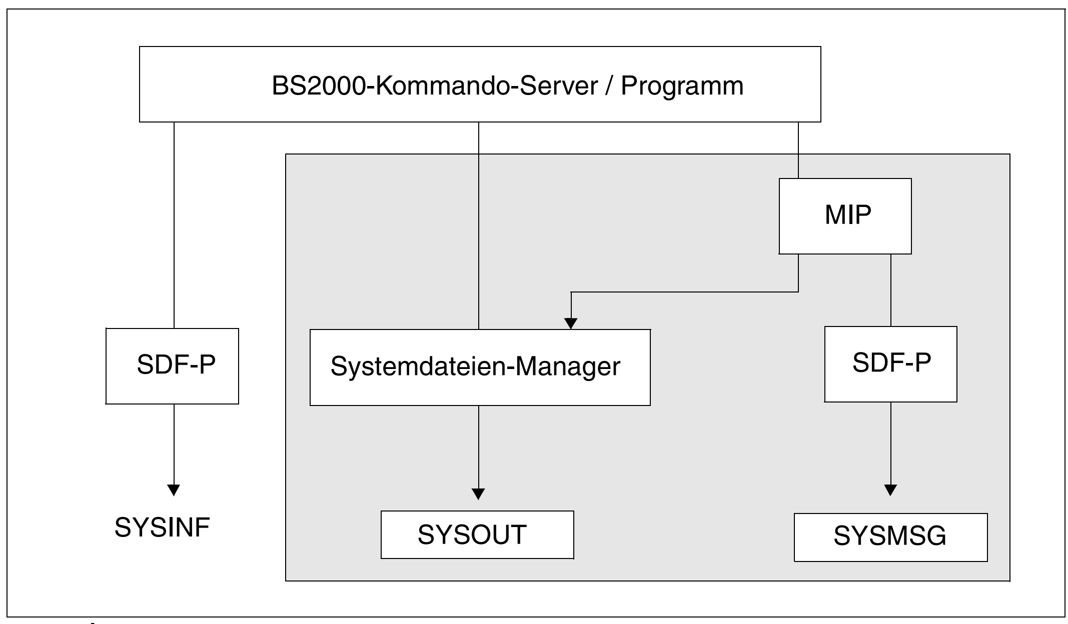



Bild 33: Ausgabe von Meldungen in den S-Variablenstrom SYSMSG und nach SYSOUT

Dem Benutzer erwachsen aus der Verwendung von S-Variablen entscheidende Vorteile:

– Die S-Variablen erlauben einen einfachen Zugriff auf eine garantierte Meldung, da die Information strukturiert abgespeichert ist. Der Anwender ist unabhängig vom Ausgabelayout der Meldung.

– Die Namen der S-Variablen und ihre inhaltliche Zuordnung sind auch in kommenden Subsystem- bzw. BS2000/OSD-Versionen garantiert.

– Der Inhalt der S-Variablen kann als Eingabe in S-Prozeduren weiterverwendet werden.

Folgende Angaben sind erforderlich, um garantierte Meldungen in eine strukturierte S-Variable auszugeben:

1. Der Anwender muss eine Listenvariable vom Typ „Struktur“ deklarieren.
   ```
   /DECLARE-VARIABLE <var-name>(TYPE=*STRUCTURE),MULTIPLE-ELEMENT=*LIST
   ```

2. Der Anwender muss den S-Variablenstrom SYSMSG für die strukturierte Ausgabe in S-Variablen zuweisen. Eine der folgenden Vereinbarungen muss getroffen werden:

   *Auf Kommandoebene:*

   – Kommando EXECUTE-CMD vereinbart die strukturierte Ausgabe in S-Variablen für ein Kommando.
   ```
   /EXEC-CMD (<cmd-name>),MSG-STRUCTURE-OUTPUT=<var-name>
   ```

   – Kommando ASSIGN-STREAM; die Zuweisung des S-Variablenstroms ist für alle nachfolgenden Kommandos gültig, solange bis die Vereinbarung wieder zurückgenommen wird `(ASSIGN-STREAM SYSMSG,TO=*STD)`.
   ```
   /ASSIGN-STREAM STREAM-NAME=SYSMSG,TO=*VARIABLE(<var-name>)
   ```

   *Auf Programmebene:*

   Der Makro CMD vereinbart die strukturierte Ausgabe in S-Variablen für mehrere Kommandos.
   ```
   CMD '<cmd-name>',...,VER=4,MSGVAR@=<adr>,MSGVARL=<len>,[,MSGEXT=YES]
   ```

   `<adr>` ist die symbolische Adresse, die den Namen der S-Variablen enthält.
   Makro CMD, siehe Handbuch „Makroaufrufe an den Ablaufteil“ [30].

3. Im Makro MSG7X muss das Ziel der Meldungsausgabe vereinbart werden.
   Wird DESTINATION=SYSOUT vereinbart, werden garantierte Meldungen immer in S-Variablen ausgegeben, vorausgesetzt der Benutzer hat eine S-Variable deklariert.
   Darüber hinaus kann dem Operanden BUFFER ein benutzereigener Speicher und BUFFUSE=EXTERNAL zugewiesen werden.
   Makro MSG7X, siehe Handbuch „Makroaufrufe an den Ablaufteil“ [30].

4. Mit MSGMAKER muss die Meldung folgendermaßen ausgezeichnet werden:

   a) Die auszugebende Meldung muss als garantiert ausgezeichnet sein (//ADD-MSG MSG-ID=...,WARRANTY=YES). Für garantierte Meldungen erzeugt MIP folgende S-Variablen:

      – MSG-ID:
        enthält den Meldungsschlüssel.

      – REPLY:
        enthält die Antwort auf die Meldung, sofern diese vom Benutzer eine Stellungnahme fordert.

      – I0 bis I29:
        Standard-Namen der Inserts; I0 bis I29 werden standardmäßig von MIP als S-Variablennamen vergeben, wenn der Benutzer mit MSGMAKER keine Insertnamen (siehe b)) vereinbart hat.

      – MSG-TEXT:
        enthält den Meldungstext inklusive aller ersetzten Inserts.

   b) Für die Inserts der auszugebenden Meldung werden Namen vereinbart (//ADD-MSG MSG-ID=...,INSERT-ATTRIBUTES=...(NAME=...)).
      Ist die Meldung nicht garantiert, wird jedoch mindestens einem Insert der auszugebenden Meldung ein Name zugewiesen, so werden folgende S-Variablen erzeugt:

      – <insert-name>:
        Mit MSGMAKER vereinbarter Name wird in Großbuchstaben umgewandelt und von MIP als S-Variablenname verwendet.

      – MSG-ID:
        enthält den Meldungsschlüssel.

      – REPLY:
        enthält die Antwort auf eine Meldung.

      – MSG-TEXT:
        enthält den Meldungstext inklusive aller ersetzten Inserts.

   **i** Die von MIP erzeugten S-Variablen sind vom Datentyp STRING.

Die endgültige, von MIP ausgegebene S-Variable setzt sich aus zwei Teilnamen zusammen:

– Name der vom Benutzer deklarierten S-Variablen: <var-name>
  (<var-name> ist der erste Teilname der S-Variablen)

– MIP liefert den zweiten Teilnamen (MSG-ID, I0 bis I29, <insert-name>, REPLY, MSG-TEXT)

Die beiden Teilnamen bilden, durch einen Punkt voneinander getrennt, die Namen der S-Variablen.

| <var-name>.MSG-ID<br><var-name>.I0 ... <var-name>.I29<br><var-name>.<insert-name><br><var-name>.REPLY<br><var-name>.MSG-TEXT |
|---|

Alle Variablen (MSG-ID, I0, ...), die es zu einer garantierten Meldung gibt, gehören einem Listenelement an. Jede weitere Meldung wird als neues Listenelement aufgenommen. Dies ist der Fall, wenn zu einem Kommando mehrere garantierte Meldungen existieren (Aufruf des Kommandos mit /EXECUTE-CMD) oder wenn mit dem Kommando /ASSIGN-STREAM allgemein die Ausgabe garantierter Meldungen in eine S-Variable umgelenkt wird.

*Beispiel*

Das folgende Beispiel zeigt die Umlenkung der Meldungsausgabe in eine S-Variable.
Die Attribute der nicht existierenden Datei „TEST“ sollen ausgegeben werden. Die dadurch verursachte Fehlermeldung DMS0533, eine garantierte Meldung, wird in die S-Variable MSG ausgegeben:

```
/DECLARE-VARIABLE NAME=MSG(TYPE=*STRUCTURE(DEFINITION=DYNAMIC)),
   MULTIPLE-ELEMENTS=*LIST ———————————————————————————————————————————  (1)
/EXECUTE-CMD CMD=(SHOW-FILE-ATTRIBUTES FILE-NAME=TEST),TEXT-OUTPUT=*NONE,
   MSG-STRUCTURE-OUTPUT=MSG(WRITE-MODE=*EXTEND) ——————————————————————  (2)
/SHOW-VAR MSG ——————————————————————————————————————————————————————  (3)
MSG(*LIST).MSG-TEXT = %  DMS0533 REQUESTED FILE NOT CATALOGED IN
   PUBSET '1SBZ'. COMMAND TERMINATED
MSG(*LIST).MSG-ID = DMS0533
MSG(*LIST).I0 = 1SBZ ———————————————————————————————————————————————  (4)
```

(1) Die S-Variable MSG wird als dynamisch erweiterbare Listenvariable deklariert.

(2) Mit dem Kommando /EXECUTE-CMD wird die Ausgabe des Kommandos /SHOW-FILE-ATTRIBUTES in die S-Variable MSG umgelenkt. Die Ausgabe nach SYSOUT wird unterdrückt.

(3) Mit SHOW-VAR MSG wird die Listenvariable MSG ausgegeben.

(4) MIP erzeugt für diese garantierte Meldung die S-Variablen MSG-TEXT, MSG-ID und I0; die Werte dieser S-Variablen bilden das erste Listenelement der S-Variablen MSG. Wird das Kommando /EXECUTE-CMD wiederholt abgesetzt, wird jede weitere Meldung als neues Listenelement angefügt, da WRITE-MODE=*EXTEND angegeben ist (siehe (2)).

# 16 Operatorfunktionen

Die Tätigkeiten, die ein Operator während des Systemlaufs auszuführen hat, stammen aus mehreren Aufgabengebieten. Zu diesen zählen z.B.:

- die Überwachung des Systemverhaltens
- die Bedienung des Datenkommunikationssystems
- die Bedienung von Peripheriegeräten

Bei einem Server mit umfangreicher Peripherie und großem Durchsatz ist es möglich, dass ein Operator die anfallende Arbeit nicht alleine bewältigen kann. In diesem Fall ist es sinnvoll, das Arbeitsaufkommen nach Aufgabengebieten getrennt auf mehrere Operator zu verteilen.

Die Arbeitsplätze, von denen die Bedienung des Systems – das sog. Operating – erfolgt, werden Bedienstationen oder Konsolen genannt. Jedem Operator sollte eine Bedienstation zur Verfügung stehen, damit sich die Operator gegenseitig nicht behindern.

Das Privileg OPERATING kann an eine beliebige Benutzerkennung vergeben werden und von einer Benutzertask dieser Kennung kann ebenfalls das Operating erfolgen.

Ursprünglich waren mit Konsolen nur Bildschirmgeräte gemeint, die – im Unterschied zu normalen Datensichtstationen – über einen Kanalanschluss verfügen. Daher ist von solchen Konsolen aus der Austausch von Nachrichten mit dem BS2000-System auch in den Betriebsphasen möglich, in denen die Datenfernübertragungskomponenten nicht zur Verfügung stehen.

Mit dem Start des Datenkommunikationssystems eröffnet das System automatisch die DCAM-Anwendung $CONSOLE. Benutzerprogramme, die sich an $CONSOLE anschließen, können auch Funktionen des Operating wahrnehmen. Sie werden deshalb ebenfalls Konsolen genannt.

Zur Unterscheidung dienen die Begriffe „physikalische Konsole“ und „logische Konsole“.

### Physikalische Konsolen

Eine „physikalische Konsole“ ist ein Konsolgerät, das direkt (als generiertes Konsolgerät) oder indirekt (über einen SKP) an einen Kanal angeschlossen ist. Physikalische Konsolen werden auch „Bedienstation“ genannt und sind Gegenstand dieses Kapitels.

In Abhängigkeit von dem Systemparameter NBCONOPI können zwei Betriebsmodi eingestellt werden: Ist eine Anmeldung (das so genannte Operator-Logon) mit einer Operator-Kennung für physikalische Konsolen erforderlich (NBCONOPI=Y), werden die Berechtigungen der physikalischen Konsolen mit Hilfe der Operator-Rollen gesteuert.
Sind keine Operator-Kennungen für physikalische Konsolen erlaubt (NBCONOPI=N), dann werden Berechtigungsschlüssel (Routing-Codes) über den Parameterservice (Parametersatz OPR) oder das ASR-Kommando erteilt.

Als Operator-Kennung wird eine beliebige Benutzerkennung mit dem Privileg OPERATING bezeichnet.



Bei NBCONOPI=Y verfügt der Operator über einen eigenen Dateiraum, eine eigene Benutzerkennung. Für RUN-Dateien erfolgt die Ergänzung des Dateinamens mit dieser Benutzerkennung und nicht mehr mit TSOS.

Wenn das Operator-Logon eingeschaltet ist (NBCONOPI=Y), dann ist nach „System Ready“ Operating von einer physikalischen Konsole nur noch nach erfolgreicher Anmeldung unter einer Kennung mit dem Privileg OPERATING möglich (mit SET-LOGON-PARAMETERS).
Die Kennung SYSOPR mit den Privilegien OPERATING und STD-PROCESSING ist immer im System vorhanden.

Vor dem Logon ist nur das Kommando SHOW-PENDING-MSG zur Anzeige offener Fragen erlaubt. Nach erfolgreicher Anmeldung hat der Operator noch keine Berechtigungen in Form von Routing-Codes. Diese besorgt er sich mit dem Kommando
REQUEST-OPERATOR-ROLE OPERATOR-ROLE=...
Mit Hilfe des Kommandos SHOW-OPERATOR-ROLE werden die der eigenen Kennung erlaubten Operator-Rollen (Operand INFORMATION=*ROUTING-CODES) bzw. die der eigenen Kennung im Moment zugewiesenen Operator-Rollen (Operand INFORMATION=*ASSIGNMENT) angezeigt.

Zur Abmeldung vom System ist das Kommando EXIT-JOB einzugeben. Es impliziert auch die Funktion des Kommandos RELEASE-OPERATOR-ROLE OPERATOR-ROLE=*ALL und führt damit zur Rückgabe aller Berechtigungsschlüssel.

### Logische Konsolen

Eine „logische Konsole“ ist ein Benutzerprogramm, in dem eine DCAM-Anwendung läuft, die mit der Systemanwendung $CONSOLE Verbindung hält. In Abhängigkeit von der Art der Anmeldung bei $CONSOLE unterscheidet man zwischen „unberechtigten“ und „berechtigten“ logischen Konsolen.

Berechtigte logische Konsolen haben – analog der MN von physikalischen Konsolen – einen eindeutigen Berechtigungsnamen, anhand dessen sie identifiziert werden können. Dabei wird nochmals zwischen „generierten Berechtigungsnamen“ und „dynamischen Berechtigungsnamen“ unterschieden.

Um zu verdeutlichen, dass logische Konsolen aus Sicht des Systems immer Benutzerprogramme sind, in denen eine DCAM-Anwendung läuft, werden als Synonym für eine (berechtigte) logische Konsole die Begriffe „berechtigtes Benutzerprogramm“ bzw. „berechtigte Anwendung“ oder auch „$CONSOLE-Anwendung“ verwendet.

Die Beschreibung der $CONSOLE-Schnittstelle für logische Konsolen erfolgt im Kapitel „Automatisierung von Operatorfunktionen“ auf Seite 663.

## 16.1 Benutzertask mit OPERATING-Privileg

Die Bedienung des Systems kann auch von Benutzertasks betrieben werden, die das Privileg OPERATING besitzen. Mit SECOS kann das Privileg OPERATING an eine beliebige Kennung vergeben werden. Die Systemkennung SYSOPR besitzt standardmäßig die Privilegien OPERATING und STD-PROCESSING.

Der von einem Systembediener-Arbeitsplatz (bisher nur Konsolen) bekannte Datenstrom lässt sich logisch in einen Dialogstrom (Dialog Stream) entsprechend der aktiven Systembedienung und einen Ereignisstrom (Event Stream) entsprechend der reaktiven Systembedienung aufteilen.

Der **Dialogstrom** ist der kommandogebundene Anteil des Datenstroms eines Systembediener-Arbeitsplatz. Er umfasst die Abwicklung des Kommandodialogs, d.h. die Eingabe von Kommandos und die Ausgabe der kommandogebundenen Meldungen sowie u. U. das Beantworten von Zusatzinformationsanforderungen der Kommandobearbeitung.
Die systeminterne SCI-Schnittstelle (Synchronous Console Interface) ermöglicht die Eingabe aller OPERATING-Kommandos aus einer Benutzertask.

Die Kommandoeingabe im Dialogstrom erfolgt synchron, d.h. die Eingabe eines Kommandos ist erst nach Beendigung des vorhergehenden möglich. Dies ist ein wesentlicher Unterschied zum Kommandodialog an Konsolen, wo die Eingabe der Kommandos asynchron erfolgt, d.h. unabhängig von der Beendigung evtl. vorausgegangener Kommandos ist Kommandoeingabe möglich. Die synchrone Arbeitsweise des Dialogstroms ermöglicht den Einsatz von SDF-P für Operating-Prozeduren in Benutzertasks mit Privileg OPERATING.

Der **Ereignisstrom** ist der systembedingte, asynchrone Anteil des Datenstroms eines Systembediener-Arbeitsplatzes. Er umfasst im Wesentlichen freilaufende Systemmeldungen (von den Konsolen bekannt), auf welche der Operator ggf. reagieren muss. Die Integration dieser Funktionalität in einer Benutzertask erfolgt im Rahmen des **Ereignisstrom-Service** (ESS, Event-Stream-Service), der auch für Benutzer ohne OPERATING-Privileg neue Möglichkeiten eröffnet. Ein wesentliches Merkmal des Ereignisstrom-Service ist die Möglichkeit, Nachrichten des Systems asynchron, also sofort nach deren Erzeugung, an einer Datensichtstation innerhalb einer Dialogtask zu präsentieren.

Zu diesem Zweck wird eine zentrale System-Ereignisstrom-Datei (System Event Stream File, SESF) angelegt, in welcher alle System-Ereignisse von der Systemtask SEST aufgezeichnet werden.
Der Name der System-Ereignisstrom-Datei ist $SYSAUDIT.SYSLOG.ESS.SYSTEM.
Sie wird beim Hochfahren des Systems angelegt und während der Systembeendigung wieder gelöscht.
Über den Systemparameter NBESSIZE lässt sich die Maximalgröße der System-Ereignisstrom-Datei einstellen. Die Voreinstellung beträgt 40000 PAM-Seiten. Die max. Aufbewahrungsdauer der aufgezeichneten Ereignisse beträgt 96 Stunden. Nach Ablauf dieser Zeit werden sie aus Platzgründen gelöscht.

System-Ereignisse sind grundsätzlich alle freilaufenden, über Routing-Code verteilten und nicht kommandogebundenen Systemmeldungen, die auch in der CONSLOG-Datei enthalten sind. Mit Hilfe des Kommandos ASSIGN-SYSEVENT können sich Benutzertasks mit dem Privileg OPERATING einen eigenen System-Ereignisstrom zuordnen. Für diese werden zusätzlich an die eigene Task gerichtete asynchrone Nachrichten und – abhängig von der im Kommando ASSIGN-SYSEVENT getroffenen Auswahl – auch Anteile des eigenen Kommandodialogs als System-Ereignisse in der System-Ereignisstrom-Datei aufgezeichnet.

Die Anzeige der aufgezeichneten Ereignisse übernimmt eine Präsentationsfunktion mit maskenorientierter Darstellung unter Nutzung des Subsystems FHS-TPR. Mit dem Kommando SHOW-SYSEVENT-LOG wird die Präsentationsfunktion des Ereignisstroms gestartet. In Benutzertasks mit Privileg OPERATING können alle im System vorhandenen System-Ereignisströme angezeigt werden. An die eigene Task gerichtete, asynchrone Nachrichten und protokollierte Anteile des Kommandodialogs werden grundsätzlich angezeigt. Der Umfang der zusätzlich dargestellten System-Ereignisse wird durch das so genannte „Betrachterprofil“ bestimmt, das vor Aufruf der Präsentationsfunktion eingestellt werden muss. Das Betrachterprofil wird durch die Gesamtheit der mit Hilfe der Operator-Rollen eingestellten Routing-Code-Menge und den lokalen Einstellungen zur Meldungsunterdrükkung, Meldungsfilterung und Meldungsbestellung bestimmt.

Die Präsentationsfunktion bietet einen dynamischen und einen statischen Modus an. Der dynamische Modus ermöglicht die unverzügliche Darstellung neu eintreffender, asynchroner Ereignisse (wie von der Bedienstation bekannt), bietet aber keine Dialogfunktion.
Der statische Modus der Präsentationsfunktion bietet einen maskengesteuerten Benutzerdialog mit Möglichkeiten zum Blättern und Positionieren innerhalb des gesamten Ereignisstroms, mit einer Suchfunktion, einer Funktion zum Beantworten offener Systemfragen und ein Hilfesystem. In Abhängigkeit der tasklokal eingestellten Meldungssprache erfolgt die Präsentation in Englisch oder Deutsch.

Zur Aufbereitung der Ausgabemasken nutzt die Präsentationsfunktion das Subsystem FHS-TPR, welches in der Regel ab „System Ready“ verfügbar ist. Der notwendige Zugriff auf die FHS-Maskenbibliotheken SYSFHS.BS2CP.<version>.E bzw. SYSFHS.BS2CP.<version>.D und SYSFHS.FHS-TPR.<version>.E bzw. SYSFHS.FHS-TPR.<version>.D wird bei der Systeminstallation ermöglicht.

Das Kommando SHOW-SYSEVENT-LOG-ATTRIBUTES erteilt Auskünfte über die Eigenschaften und Attribute der zugreifbaren Ereignisströme.

Beliebige Benutzertasks können sich eigene Benutzer-Ereignisströme (User Event Stream) zuordnen, die keinerlei Operating-Funktionalität enthalten.
Für diese erfolgt allein die Aufzeichnung der mittels Kommando ASSIGN-SYSEVENT eingestellten Ereignisse in einer privaten Benutzer-Ereignisstrom-Datei (UESF, User Event Stream File). Die Datei wird unter der eigenen Kennung angelegt und hat den Namen SYSLOG.ESS.USER.<log-id>.

Benutzer-Ereignisströme eignen sich vor allem zum Aufzeichnen und Darstellen von asynchronen Nachrichten an die Benutzer (z.B. INFORM-JOB bzw. INFORM-ALL-JOBS) und zum Überwachen von Abläufen, die sich hinter einem Auftragskommando (z.B. ENTER-JOB) verbergen. Die Präsentationsfunktion wird ebenfalls mit dem Kommando SHOW-SYSEVENT-LOG gestartet. Alle vorhandenen Benutzer-Ereignisströme der eigenen Kennung können angezeigt werden.

Bei der Systembedienung von Konsolen ist der Operator an eine gemischte Darstellung von Dialogstrom und Ereignisstrom gewöhnt. Bei der Systembedienung aus einer Benutzertask ist diese Mischform der Darstellung aus technischen Gründen nicht möglich.
Zur Realisierung eines komfortablen Arbeitsplatzes sind mindestens zwei Dialogtasks an zwei Terminals oder – innerhalb einer Terminal-Emulation – die Mehrfenstertechnik zu empfehlen. Eine Dialogtask dient der Bedienung des Dialogstroms, die andere Dialogtask der Präsentation des Ereignisstroms. Sind weitere Dialogtasks bzw. Präsentationsfenster verfügbar, können mehrere Präsentationsfunktionen mit unterschiedlichen Betrachterprofilen entsprechend unterschiedlichen Arbeitsgebieten parallel dargestellt werden. Steht nur eine einzige Dialogtask zur Verfügung, ist trotzdem Systembedienung aus der Benutzertask möglich, nur ist in diesem Fall das Wechseln zwischen Dialogstrom und Ereignisstrom (Starten und Beenden der Präsentationsfunktion je nach Bedarf) notwendig.

### 16.1.1 Bereitstellung einer Kennung für das Operating (mit SECOS)

Zur Bereitstellung der Operating-Funktionalität von physikalischen Konsolen (bei eingeschaltetem Operator-Logon), $CONSOLE-Anwendungen und aus Benutzertasks sind die im Folgenden beschriebenen Vorbereitungen zu treffen. Zur Nutzung der vorhandenen Systemkennung SYSOPR ist nur das Einrichten und Zuordnen von Operator-Rollen notwendig.

*Unter der Kennung TSOS*

- Einrichten einer Kennung:
  ```
  /ADD-USER USER-ID=OPEROPER, LOGON-PASSWORD=C'12345678', ACCOUNT=account
  ```
- Für $CONSOLE-Anwendungen kann u.U. noch ein erweiterter Zugangsschutz mit folgendem Kommando festgelegt werden:
  ```
  /MODIFY-LOGON-PROTECTION USER-ID=OPEROPER, ...,
                           OPERATOR-ACCESS-TERM= ...,
                           OPERATOR-ACCESS-PROG= ...
  ```

*Unter der Kennung SYSPRIV*

- Erteilen des Privilegs OPERATING (für $CONSOLE-Anwendungen nicht unbedingt erforderlich):
  ```
  /SET-PRIVILEGE PRIVILEGE=OPERATING, USER-ID=OPEROPER
  ```
- Einrichten von Operator-Rollen:
  ```
  /CREATE-OPERATOR-ROLE OPERATOR-ROLE=OPERROLE, ROUTING-CODES=<list>
  ```
- Zuordnen der Operator-Rollen zu der Kennung:
  ```
  /MODIFY-OPERATOR-ATTRIBUTES USER-ID=OPEROPER,ADD-OPERATOR-ROLE=OPERROLE
  ```

Nach diesen Vorbereitungen und der Eingabe des Kommandos SET-LOGON-PARAMETERS mit Benutzerkennung, Abrechnungsnummer und Kennwort ist es für einen Operator unter Berücksichtigung der jeweiligen Besonderheiten möglich, von einer physikalischen Konsole, von einer $CONSOLE-Anwendung oder von einer Benutzertask aus Systembedienung zu betreiben. Die Berechtigungsschlüssel fordert der Operator mit dem Kommando REQUEST-OPERATOR-ROLE an.

### 16.1.2 Nutzung des Ereignisstrom-Service für Operating aus Benutzertasks

Die Nutzung für einen komfortablen Operator-Arbeitsplatz basiert auf mindestens zwei Benutzertasks. Eine Benutzertask (Benutzertask-1) wird zur Eingabe von Kommandos und zum Empfang der zugehörigen Kommandoergebnisse verwendet (Dialogstrom). Die zweite Benutzertask (Dialogtask-2) ist notwendigerweise eine Dialogtask und dient der Präsentation der asynchronen Systemmeldungen (Ereignisstrom).

**Dialogstrom (Benutzertask-1):**

- Die Eingabe aller mit dem Privileg OPERATING geschützten Kommandos ist erlaubt.

  Die Berechtigungsprüfung erfolgt anhand der SDF-Syntax-Dateien (einschließlich einer evtl. vorgeschaltenen Gruppen-Syntax-Datei).

  Bei Kommandos, die über die interne SCI-Schnittstelle der UCON-Task in einer Operator-Task (Umgebung) abgewickelt werden, erfolgt keine Berechtigungsprüfung anhand der Authorization-Codes in den Kommandotabellen der UCON-Task.
  Ausnahme: Mit dem Authorization-Code „$“ geschützte Kommandos sind auch aus Benutzertasks verboten.

- Einrichten eines System-Ereignisstroms zum Empfang asynchroner Systemmeldungen:
  ```
  /ASSIGN-SYSEVENT TO=*SYSTEM (LOG-ID=OPER),ADD-SYNCH-EVENTS=*NONE
  ```
  Mit ADD-SYNCH-EVENTS=*ALL kann zusätzlich der gesamte Kommandodialog der Benutzertask-1 im Ereignisstrom OPER protokolliert werden. Mit dem Kommando ASSIGN-SYSEVENT ..., ADD-SYNCH-EVENTS wird der Umfang der zu protokollierenden Ereignisse (Nachrichten) eingestellt.

  Die an die Benutzertask-1 direkt gesendeten asynchronen Nachrichten werden grundsätzlich protokolliert. Je nach Einstellung (Kommando MODIFY-MSG-OPTIONS) werden diese zusätzlich auch an der Datensichtstation ausgegeben. Die Ausgabe an der Datensichtstation kann durch folgende Einstellung unterdrückt werden:
  ```
  /MODIFY-MSG-OPTIONS OPERATOR-BROADCAST=*NO,OPERATOR-MSG=*NO,SYSTEM-MSG=*NO
  ```

- Die Präsentation des Ereignisstroms in derselben Task ist möglich, aber nicht empfehlenswert, da während der Präsentation keine Kommandoeingabe möglich ist.

- Die Präsentation des Ereignisstroms erfolgt aus der zentralen System-Ereignisstrom-Datei. Diese enthält – im Rahmen der durch den Systemparameter NBESSIZE festgelegten Größenordnung – alle notwendigen (bei der Präsentation auswählbaren) Nachrichten der letzten Stunden des Systemlaufs (max. 96 Stunden).

**Präsentation des Ereignisstroms (Dialogtask-2):**

- Die Dialogtask-2, ebenfalls unter einer Kennung mit Privileg OPERATING, dient nur der Präsentation des Ereignisstroms. Der in Benutzertask-1 zugewiesene Ereignisstrom ist für Dialogtasks aller Kennungen mit Privileg OPERATING zugreifbar und anhand seines Namens (LOG-ID) eindeutig identifizierbar.
- Für die Präsentation werden aus der Gesamtheit der aufgezeichneten Ereignisse über die tasklokale Einstellung des Betrachterprofils die darzustellenden Ereignisse anhand der nachfolgenden Kriterien ausgewählt. Die Aufzählung erfolgt in der Reihenfolge ihrer Priorität, d.h. sofern ein Kriterium erfüllt ist, werden die folgenden Ereignisse nicht mehr geprüft:

  1. Protokollierte Anteile des Kommandodialogs von Benutzertask-1 werden immer dargestellt.
  2. Direkt an die Benutzertask-1 gesendete, asynchrone Nachrichten werden dargestellt.
  3. Anhand ihrer MSG-ID unterdrückte System-Nachrichten (Kommandos SET- und RESET-MSG-SUPPRESSION) werden nicht dargestellt, sofern es keine Fragen sind.
  4. Anhand ihrer MSG-ID bestellte System-Nachrichten (Kommando MODIFY-MSG-SUBSCRIPTION) werden dargestellt.
  5. Mit Hilfe der eingestellten Operator-Rollen (Kommandos REQUEST- und RELEASE-OPERATOR-ROLE) werden über Routing-Code verteilte Systemmeldungen zur Darstellung ausgewählt.
  6. Bei über Routing-Code verteilten Meldungen werden die eingestellten Routing-Code-spezifischen Filterstufen (Kommandos ADD- und REMOVE-CONSOLE-FILTER) gegen das Meldungsgewicht geprüft.
  7. Ereignisstrom-spezifische Meldungen (z.B. erfolgreich durchgeführte Recovery oder Reorganisation) werden unabhängig von taskspezifischen Einstellungen immer angezeigt.

- Gestartet wird die Präsentationsfunktion mit dem Kommando
  `/SHOW-SYSEVENT-LOG LOG-ID=OPER(TYPE=*SYSTEM)`

  Die Präsentationsfunktion meldet sich mit ihrer Hauptmaske im statischen Modus und bietet Eingabemöglichkeiten über Menü, Kommandozeile und F-Tasten. Die aktuelle Maske positioniert immer auf die jüngsten Ereignisse.

  Folgende Funktionen werden im statischen Modus angeboten:

  – Positionieren und Blättern innerhalb des Ereignisstroms
  – Suchen
  – Beantworten offener Systemfragen
  – Wechseln in den dynamischen Modus
  – Bereitstellen eines Hilfesystems

  Der dynamische Modus bietet die für die Systembedienung wichtigste Funktionalität: die sofortige Ausgabe asynchron eintreffender Ereignisse wie von der Konsole bekannt. Die einzig mögliche Eingabe ist die K2-Taste zur Rückkehr in den statischen Modus.

Wird nur die Präsentation der System-Ereignisse ohne „Privatanteile" gewünscht, ist das Einrichten eines System-Ereignisstroms mit dem Kommando ASSIGN-SYSEVENT nicht erforderlich. Der direkte Aufruf der Präsentationsfunktion mit
`/SHOW-SYSEVENT-LOG LOG-ID=*SYSTEM-MSG-ONLY`

zeigt die dem „Betrachterprofil" des Operators entsprechende Auswahl an System-Ereignissen.

Für eine ausführliche Beschreibung der Präsentationsfunktion wird auf die Beschreibung des Kommandos SHOW-SYSEVENT-LOG verwiesen.

### 16.1.3 Nutzung des Ereignisstrom-Service für beliebige Benutzertasks

Die Funktionalität des Ereignisstrom-Service ist für jede beliebige Benutzertask nutzbar, jedoch um die spezifische Operating-Funktionalität eingeschränkt. Es ist sinnvoll, pro Arbeitsplatz mindestens zwei Benutzertasks zu vewenden. Eine Benutzertask (Benutzertask-1) wird zur Eingabe von Kommandos und zum Empfang der zugehörigen Kommandoergebnisse verwendet. Die zweite Benutzertask (Dialogtask-2) ist notwendigerweise eine Dialogtask und dient der Präsentation asynchroner, an die Benutzertask gerichteter Nachrichten (z.B. INFORM-ALL-JOBS), sowie von Anteilen des Kommandodialogs von Benutzertask-1 je nach Einstellung.

Für jeden Benutzer-Ereignisstrom erfolgt die Aufzeichnung der Ereignisse in einer eigenen Benutzer-Ereignisstrom-Datei in einer eigenen Batchtask der gleichen Kennung. Der Name der Datei ist SYSLOG.ESS.USER.log-id. Den Suffix des Dateinamens bildet der Name (LOG-ID) des Ereignisstroms. Der Dateizugriff ist nur über die Präsentationsfunktion möglich. Die Lebensdauer der Batchtask entspricht der aktiven Phase des Ereignisstroms (vom Starten bis zum Beenden bzw. Schließen). Die Benutzer-Ereignisstrom-Datei wird beim erstmaligen Starten des Ereignisstroms erzeugt und beim endgültigen Beenden gelöscht.

Der Name des Ereignisstroms (LOG-ID) ist innerhalb der Kennung eindeutig. Der Zugriff auf einen Benutzer-Ereignisstrom ist von allen Dialogtasks der gleichen Kennung möglich. Es gibt keine Sonderberechtigung für die Kennung TSOS.

**Dialogstrom (Benutzertask-1):**

- Einrichten eines Benutzer-Ereignisstroms zum Empfang asynchroner Nachrichten:
  `/ASSIGN-SYSEVENT TO=*USER (LOG-ID=MYES), ADD-SYNCH-EVENTS=*NONE`

  Mit ADD-SYNCH-EVENTS=*ALL kann zusätzlich der gesamte Kommandodialog der Benutzertask-1 im Ereignisstrom MYES protokolliert werden. D.h. mit Hilfe des Kommandos ASSIGN-SYSEVENT und dem Operanden ADD-SYNCH-EVENTS wird der Umfang der zu protokollierenden Ereignisse (Nachrichten) eingestellt.

  Die an die Benutzertask-1 direkt gesendeten asynchronen Nachrichten werden grundsätzlich protokolliert. Je nach Einstellung (Kommando MODIFY-MSG-OPTIONS) werden diese zusätzlich auch an der Datensichtstation ausgegeben. Die Ausgabe an der Datensichtstation kann durch folgende Einstellung unterdrückt werden.
  `/MODIFY-MSG-OPTIONS OPERATOR-BROADCAST=*NO.OPERATOR-MSG=*NO,SYSTEM-MSG=*NO`

- Die Präsentation des Ereignisstroms in derselben Task ist möglich, aber nicht empfehlenswert, da während der Präsentation keine Kommandoeingabe möglich ist.

**Präsentation des Ereignisstroms (Dialogtask-2):**

- Die Dialogtask-2, ebenfalls unter der gleichen Kennung dient nur der Präsentation des Ereignisstroms. Der in Benutzertask-1 zugewiesene Ereignisstrom ist anhand seines Namens (LOG-ID) eindeutig identifizierbar.
- Es werden immer alle aufgezeichneten Ereignisse angezeigt, d.h. es gibt kein „Betrachterprofil“ für Benutzer-Ereignisströme.
- Gestartet wird die Präsentationsfunktion mit dem Kommando
  `/SHOW-SYSEVENT-LOG LOG-ID=MYES`

  Die Präsentationsfunktion meldet sich mit ihrer Hauptmaske im statischen Modus und bietet Eingabemöglichkeiten über Menü, Kommandozeile und F-Tasten. Die aktuelle Maske positioniert immer auf die jüngsten Ereignisse.

  Folgende Funktionen werden im statischen Modus angeboten:

  – Positionieren und Blättern innerhalb des Ereignisstroms
  – Suchen
  – Wechseln in den dynamischen Modus
  – Bereitstellen eines Hilfesystems

Für eine ausführliche Beschreibung der Präsentationsfunktion wird auf die Beschreibung des Kommandos SHOW-SYSEVENT-LOG verwiesen.

## 16.2 Kommandoeingabe über Bedienstation

Die Tastaturfelder sind bei den verschiedenen Server-Typen unterschiedlich.

### Eingabe der Kommandos

1. EINGEB-Taste drücken
2. Eingabe eines Schrägstriches (/), dem das Kommando und die erforderlichen Operanden folgen
3. ETX-Taste (DÜ1-Taste oder ENTER-Taste) drücken

### Formate der Kommandos

```
/[.cid] cmd␣operands
```

| | |
|---|---|
| . | Punkt |
| cid | Kommandoauftragskennzeichen<br>Das cid beginnt mit einem Buchstaben oder einem der Zeichen @, $, #, an das sich noch 7 Zeichen (A-Z, 0-9, oder @,$,#) anschließen können.<br>Es entspricht formal den Job-Namen bei Benutzerkommandos (siehe Handbuch „Kommandos“ [27]).<br>Das cid dient der Auftragsklammerung bei Operatorkommandos.<br>Die ersten drei Zeichen des cid erscheinen als mid (Meldungsauftragskennzeichen, siehe „Format der Meldungen“, Seite 624) in der Kommandobeendigungsmeldung, die vom System für alle Operatorkommandos gesendet wird und in allen Kommandoergebnissen. |
| cmd | Bezeichnung eines Operator- oder Spezialkommandos |
| ␣ | Leerzeichen |
| operands | Operanden des Kommandos |

Die Kommandos können abgekürzt werden, solange sie eindeutig bleiben.

*Beispiel*

```
1. /.MINE CANCEL-JOB *TSN(1532)
2. /.YOURS SPECIAL OP1=<value1>,OP2=<value2>
```

Das Kommando unter 1. ist ein gewöhnliches Operatorkommando,
das Kommando unter 2. ist ein Spezialkommando.

# 16.3 Nachrichten

## 16.3.1 Emergency-Nachrichten

Schwerwiegende Systemprobleme, wie z.B. Speichermangel oder Hardware-Fehler bei Systemplatten, können die BS2000-Komponenten treffen, die für die Ein-/Ausgabe von Konsolnachrichten zuständig sind. In Situationen, in denen die normalen Ausgabewege blockiert sind, wählt das System für wichtige Nachrichten einen alternativen Ausgabeweg, den sog. „Emergency-Pfad“.

Ausgaben über den Emergency-Pfad werden Emergency-Meldungen bzw. Emergency-Fragen genannt. Sie werden zunächst über interne Service-Schnittstellen direkt an der physikalischen Hauptkonsole des Systems ausgegeben; an anderen physikalischen Konsolen ist die Ausgabe server-abhängig.

#### Formate

| | |
|---|---|
| %E.text | Emergency-Meldung |
| ?E.text | Emergency-Frage |

Anschließend werden – wenn möglich – die Nachrichten der regulären Nachrichtenverarbeitung übergeben, siehe die folgenden Abschnitte.

Unabhängig von der Ausgabe an logischen Konsolen oder der Anzeige durch das Kommando SHOW-PENDING-MSG ist das Beantworten von Emergency-Fragen nur an einer physikalischen Konsole möglich, an der die Meldung ausgegeben wurde.

#### Format der Antwort

| | |
|---|---|
| E.text | Antwort auf Emergency-Frage |

## 16.3.2 Verständigung zwischen System und Operator

Das System und seine Benutzer können Nachrichten an die Bedienstationen senden und Antworten von dort erhalten, wobei der „Benutzer“ auch ein Operator sein kann, der selbst an einer Bedienstation sitzt. Inhalt und Bedeutung dieser Nachrichten sind naturgemäß sehr verschieden. Um entscheiden zu können, welche Nachrichten wie behandelt werden müssen, unterscheidet man sie nach Typen und Eigenschaften.

### Nachrichtentypen

Um dem Empfänger möglichst schnell einen Anhaltspunkt über Art und Wichtigkeit der Nachrichten zu geben, stellt das System eine Typeinteilung bereit. Jede auf der Konsole ausgegebene Nachricht wird genau einem dieser Typen zugeordnet, und jeder dieser Typen hat ein eindeutiges Typkennzeichen, das mit ausgegeben wird.

Folgende Nachrichtentypen sind definiert:

% „Meldung“: Informative Nachricht zur Kenntnisnahme durch den Operator.

? „Frage“: Nachricht, in der zur Eingabe einer Antwort aufgefordert wird.

. „Antwort“: Antwort des Operators auf eine Frage.

; „Anweisung“: Nachricht, mit der der Operator aufgefordert wird, etwas zu tun. Aus der internen Sicht des Systems handelt es sich bei Anweisungen ebenfalls um Fragen, jedoch mit dem Unterschied, dass die Antwort nicht vom Operator gegeben werden kann, sondern nur vom Absender der Frage selbst.

/ „Kommando“: Typkennzeichen für die Eingabe eines Kommandos.

+ „Kommandoergebnis“: Nachricht eines Kommandoservers an den Absender des Kommandos.

& „Kommandozusatzinformationsanforderung“: Nachricht, in der der Absender eines Kommandos zur Eingabe weiterer Informationen aufgefordert wird.

: „Kommandozusatzinformation“: Antwort des Kommandogebers auf eine Kommandozusatzinformationsanforderung.

! „Kommandoende“: Nachricht, mit der dem Eingeber eines Kommandos das Ende der Ablaufüberwachung durch das System mitgeteilt wird. Im Normalfall ist dies identisch mit der Beendigung des Kommandos, es gibt jedoch Ausnahmen. Näheres siehe Abschnitt „Kommandobeendigungsmeldungen“ auf Seite 655.

* „Rückweisung“: Nachricht, mit der UCON eine fehlerhafte Eingabe zurückweist. Alle möglichen Rückweisungen sind im Abschnitt „Rückmeldungen in berechtigten Benutzerprogrammen und an Bedienstationen“ auf Seite 658 zusammengestellt.

### Nachrichteneigenschaften

Für die Nachrichtenbehandlungsmechanismen und -kommandos des Systems sind ferner einige weitere, nicht immer aus dem Typ ableitbare Eigenschaften einer Nachricht entscheidend. Es werden unterschieden:

- „beantwortbare“ und „unbeantwortbare“ Nachrichten
- „kommandogebundene“ und „kommandoungebundene“ Nachrichten
- „gerichtete“ und „ungerichtete“ Nachrichten

*Beantwortbare und unbeantwortbare Nachrichten*

„Beantwortbar“ sind alle Nachrichten der Typen:

- Frage
- Kommandozusatzinformationsanforderung
- Anweisung

„Unbeantwortbar“ sind alle Nachrichten anderer Typen.

Wichtig ist diese Nachrichteneigenschaft z.B. bei der Ausgabe, wo beantwortbare Meldungen hervorgehoben (auf Farbbildschirmen z.B. rot anstatt grün) erscheinen und mit dem Kommando SHOW-PENDING-MSG wieder angezeigt werden können.

*Kommandogebundene und kommandoungebundene Nachrichten*

„Kommandogebunden“ sind – neben dem Kommando selbst – alle Nachrichten der Typen

- Kommandoergebnis
- Kommandozusatzinformationsanforderung
- Kommandozusatzinformation
- Kommandoende

„Kommandoungebunden“ sind alle Nachrichten anderer Typen.

Wichtig ist diese Nachrichteneigenschaft bei der Programmierung von Kommandoservern für Operator-Spezialkommandos (siehe Seite 684).

*Gerichtete und ungerichtete Nachrichten*

„Gerichtete“ Nachrichten sind an einen exakt definierten Empfänger gerichtet.

„Ungerichtete“ Nachrichten haben dagegen einen Routing-Code als Empfängerangabe. Sie werden vom System an alle Konsolen verteilt, die den angegebenen Routing-Code innehaben – das können sehr viele sein, aber auch gar keine.

Im Gegensatz zu den anderen Nachrichteneigenschaften ist der Nachrichtentyp hier kein eindeutiges Kriterium: Zwar sind z.B. Antworten (an den Fragesteller) und alle kommandogebundenen Nachrichten grundsätzlich gerichtet, die anderen Nachrichtentypen können jedoch sowohl gerichtet als auch ungerichtet sein. Zuverlässig erkennen lässt sich das nur in der CONSLOG, wo der Empfänger jeder Nachricht mitprotokolliert wird, und in berechtigten Benutzerprogrammen, die ihre Nachrichten in erweiterter Form empfangen (siehe Seite 663).

Eine Sonderbehandlung gilt hier für Emergency-Nachrichten: Aus technischen Gründen gehen sie zunächst gerichtet an die Hauptkonsole. Anschließend werden sie, sofern noch möglich, ungerichtet über den Routing-Code * verteilt, jedoch nur an logische, nicht an physikalische Konsolen.

Wichtig ist diese Nachrichteneigenschaft z.B. bei der Meldungsfilterung (Kommandos ADD-/REMOVE-CONSOLE-FILTER): Dieser Mechanismus wirkt nur auf ungerichtete Meldungen mit dem Nachrichtentyp „%“ (Meldung).

### Ausgabeformat

Alle Nachrichten werden bei der Ausgabe auf Konsole mit einem einheitlichen, 17 Zeichen langen Vorspann versehen.
Ausnahme: Antworten auf Fragen des Operators erhalten keine Uhrzeit-Angabe.

Spalte 1 Typkennzeichen der Nachricht: mögliche Werte siehe Seite 619;
Ausnahme: bei Emergency-Nachrichten nur „?“ und „Y“.

Spalte 2-5 Absender der Nachricht; möglich Notationen:

(mn) Absender ist die physikalische Konsole mit der MN „mn“
name Absender ist die logische Konsole mit dem Berechtigungsnamen „name“
tsn Absender ist die Task mit der TSN „tsn“

Spalte 6 Bindestrich (Minuszeichen)

Spalte 7-9 Meldungsauftragskennzeichen:
3-stelliges, alphanumerisches Kennzeichen der Nachricht

Spalte 10 Trennzeichen:
Ist der Systemparameter SECSTART=N, so ist das Trennzeichen immer ein Punkt.
Ist SECSTART=Y, so ist das Trennzeichen dann ein „#“, wenn die Nachricht vom System selbst stammt, also „vertrauenswürdig“ ist. Andere Nachrichten, z.B. vom Benutzer mit INFORM-OPERATOR ausgegebene, erhalten als Trennzeichen einen Punkt.

Spalte 11-16 Uhrzeit im Format: hhmmss
(bei Antworten folgt ab Spalte 11 der Antworttext)

Spalte 17 Leerzeichen

Spalte 18ff Nachrichtentext

*Beispiel*

```
% 6C5-000.122342 % BLS0519 PROGRAM 'EDT' LOADED
```

### 16.3.3 Nachrichtenformate

Folgende Formate werden beschrieben:

- Formate beim Empfangen von Meldungen im Dialog zwischen System und Benutzer
- Format beim Senden von Antworten
- Format der kommandogebundenen Antworten
- Format bei der Eingabe von Operatorkommandos
- Format der Kommandoergebnisse und Kommandobeendigungsmeldungen

### Formate beim Empfangen von Meldungen im Dialog zwischen System und Benutzer

```
{?}                
{%} tsn-mid {.} hhmmss␣text
{;}         {#}
```

| | |
|---|---|
| % | Meldung, Antwort nicht erforderlich |
| ? | Meldung, die eine Antwort verlangt, die auch vom Operator gegeben werden kann |
| ; | Meldung, die eine Antwort verlangt, die nicht vom Operator gegeben werden kann |
| tsn | Auftragsnummer der Task, die die Systemmeldung verursacht hat |
| - | Bindestrich |
| mid | Meldungsauftragskennzeichen, Standardwert = 000 (A..Z, 0-9, @, #, $) |
| . | Punkt als Trennzeichen. Der Sender ist ein Benutzerprogramm. |
| # | Nummernzeichen als Trennzeichen. Der Sender ist ein Systemmodul. Ob Punkt oder „#“ ist abhängig vom Systemparameter SECSTART (siehe Anhang, Seite 757). |
| hhmmss | Tageszeit (6 Ziffern; hh = Stunde, mm = Minute, ss = Sekunde) |
| text | Beliebiger Meldungstext |

*Beispiel*

```
%TSN1-000#142423 % NBR0970 OPERATOR TASK WITH TSN 'XAAA' CREATED FOR
                   CONSOLE 'K1'
?TSN9-000#142753 % NBR0820 SHOULD K1 BECOME MAIN OPERATOR TERMINAL ?
;TSN4-001#142823 % XYZ0123 SWITCH OFF PRINTER L1 URGENTLY
```

Meldungen des Typs „?“ und „;“ können mit dem Kommando SHOW-PENDING-MSG wieder angezeigt werden.
Meldungen des Typs „;“ können nur vom Meldungssender wieder aus der Warteschlange der offenen Fragen entfernt werden.
Der Operator hat nur indirekt Einfluss darauf, indem er die vom Sender verlangte Reaktion durchführt (im obigen Beispiel den Drucker L1 abschaltet).

### Format beim Senden von Antworten

```
dest [-mid].[text]
```

dest Ziel der Antwort. Dies ist der Absender der zu beantwortenden Meldung.

\- Bindestrich

mid Meldungsauftragskennzeichen (A..Z, 0-9, @, #, $); führende Nullen dürfen entfallen (mid muss der Meldung, die beantwortet werden soll, entsprechen)

. Punkt, Kennzeichen für Antwort

text Beliebiger Antworttext
Bei Antworten auf Systemmeldungen sind nur bestimmte Texte zulässig.
Hilfe zu einzelnen Meldungen erhalten Sie mit /HELP-MSG-INFORMATION.

*Beispiel*

```
TSN9-000.YES
TSN9.YES
```

### Format der kommandogebundenen Antworten

```
dest [-mid]:[text]
```

dest Ziel der Kommandozusatzinformation
Dies ist der Absender der zu beantwortenden Meldung.

\- Bindestrich

mid Meldungsauftragskennzeichen (A..Z, 0-9, @, #, $)
(muss der Meldung, die beantwortet werden soll, entsprechen)

: Doppelpunkt, Kennzeichen für Kommandozusatzinformation

text Beliebiger Antworttext
Bei Antworten auf Systemmeldungen sind nur bestimmte Texte zulässig.
Hilfe zu einzelnen Meldungen erhalten Sie mit /HELP-MSG-INFORMATION.

*Beispiel*

```
TSN3-TUR:Y
```

### Format bei der Eingabe von Operatorkommandos

```
/[.cid] ␣ cmd␣ [operands]
```

/ Schrägstrich. Kennzeichen für Kommando

. Punkt. Kennzeichen, dass ein „cid“ angegeben ist

cid Kommandoauftragskennzeichen
Das cid beginnt mit einem Buchstaben oder einem der Zeichen @, $, #, an das sich noch 7 Zeichen (A-Z, 0-9, oder @,$,#) anschließen können. Es entspricht formal den Job-Namen bei Benutzerkommandos (siehe Handbuch „Kommandos“ [27]).
Das cid dient der Auftragsklammerung bei Operatorkommandos. Die ersten drei Zeichen des cid erscheinen als mid (Meldungsauftragskennzeichen, siehe „Format von Meldungen“, Seite 624) in allen kommandogebundenen Nachrichten (Kommandoergebnisse, Zusatzinfo-Anforderungen und Zusatzinfos).

cmd Operationsbezeichnung eines Operatorkommandos

␣ Leerzeichen

operands
Operanden des Kommandos

*Beispiele*

```
/.MINE CANCEL-JOB *TSN(1532)
/SPECIAL-CMD OP1=<value1>,OP2=<value2>
/.EVAN16 SHOW-USER-ATTRIBUTES USER-IDENTIFICATION=RZ01
/.E SHOW-USER-ATTRIBUTES USER-IDENTIFICATION=RZ01
```



Im Gegensatz zum Meldungsauftragskennzeichen werden bei der Eingabe von Kommandoauftragskennzeichen nur bis zu acht Zeichen zugelassen. Ein- oder zweistellige Kommandoauftragskennzeichen werden auf bis zu drei Stellen nach rechts mit Nullen ergänzt. Kommandoauftragskennzeichen müssen mit einen Buchstaben oder den Sonderzeichen $,#,@ beginnen. Die ersten drei Zeichen des Kommandoauftragskennzeichens werden als Meldungsauftragskennzeichen verwendet.

### Format der Kommandoergebnisse und Kommandobeendigungsmeldungen

```
 ⎧+⎫ ⎧tsn⎫       ⎧.⎫
⎨&⎬ ⎨   ⎬ -mid ⎨ ⎬hhmmss text
 ⎩!⎭ ⎩an ⎭       ⎩#⎭
```

+      Kommandoergebnis, Antwort nicht erforderlich

&      Kommandozusatzinformationanforderung, der Absender verlangt eine Antwort

!      Kommandoende, Antwort nicht erforderlich

tsn      Auftragsnummer der Task, in der die Kommandobearbeitung stattfindet

an      Berechtigungsname des berechtigten Benutzerprogramms, das die Kommando-bearbeitung durchführt (Operator-Spezialkommando)

-      Bindestrich

mid      Meldungsauftragskennzeichen, Standardwert = 000 (A..Z, 0-9, @, #, $)

.      Punkt als Trennzeichen

#      Nummernzeichen als Trennzeichen
Ob als Trennzeichen der Punkt oder das Zeichen „#“ verwendet wird, ist abhängig vom Systemparameter SECSTART (siehe Anhang, Seite 757).

hhmmss
Tageszeit (6 Ziffern; hh = Stunde, mm = Minute, ss = Sekunde).

text      Beliebiger Meldungstext

*Beispiele*

```
&TSN3-TUR.142623  %  EXC0422 PROCESSING OF /TURN TO BE CONTINUED? REPLY
                             (Y=YES; N=NO)

+TSN2-ASR.142523  %  NBR0825 THIS CONSOLE IS (K3)

!TSN2-ASR.142723  %  NBR0740 COMMAND COMPLETED 'ASR' (RESULT: SC2=00,
                             SC1=00, MC=CMD0001); DATE :<date>
```

## 16.3.4 Steuerung der Nachrichtenzustellung

Die Nachrichten, die an eine Bedienstation gesendet werden können, lassen sich nach mehreren Kriterien unterscheiden (siehe auch Abschnitt „Nachrichten“ auf Seite 618).

Um die Zustellung der Nachrichten zu steuern, stehen derzeit vier Mechanismen zur Verfügung:

- NOINF-Funktion
- Nachrichtenunterdrückung durch Filterung
- Meldungsbestellung anhand der Meldungsschlüssel
- Nachrichtenunterdrückung anhand der Meldungschlüssel

i Auf das Konsollogging (Funktion CONSLOG) hat keiner der vier Mechanismen Einfluss.

Für die Funktionen zur Nachrichtenunterdrückung gilt: Beantwortbare Nachrichten sind grundsätzlich nicht unterdrückbar.

| **Kommando** | **Bedeutung** |
|---|---|
| ADD-CONSOLE-FILTER | Ausgabe bestimmter Meldungsgruppen unterdrücken oder Konsole in den NOINF-Zustand versetzen |
| ASR | Konsole in den NOINF-Zustand versetzen |
| MODIFY-MSG-SUBSCRIPTION | Meldungsbestellung verfügen oder zurücknehmen |
| REMOVE-CONSOLE-FILTER | Mit ADD-CONSOLE-FILTER eingestellte Filterstufen aufheben |
| RESET-MSG-SUPPRESSION | Unterdrückung der Meldungsausgabe aufheben |
| SET-MSG-SUPPRESSION | Gezielt die Ausgabe einzelner Meldungen auf eine Bedienstation oder ein berechtigtes Benutzerprogramm unterdrücken |
| SHOW-CONSOLE-FILTER | Einstellung der Filterstufen anzeigen |
| SHOW-MSG-SUBSCRIPTION | Meldungsbestellung und Unterdrückung nicht bestellter Meldungen anzeigen |
| SHOW-MSG-SUPPRESSION | Informationen über Meldungsunterdrückung anfordern |
| **OPR-Anweisung** | **Bedeutung** |
| SET-FILTER | Ausgabe bestimmter Meldungsgruppen unterdrücken;<br>Filterstufen einstellen |

Tabelle 47: Schnittstellenübersicht zur Steuerung der Nachrichtenzustellung

Der Empfängerkreis einer Meldung wird auf folgende Weise ermittelt:

- Wenn es eine an eine bestimmte Bedienstation gerichtete Meldung ist, so wird diese Bedienstation in den Empfängerkreis der Meldung eingetragen.
- Wenn es eine über einen Routing-Code verteilte (d.h. ungerichtete) Meldung ist, dann werden alle Bedienstationen in den Empfängerkreis eingetragen, die Inhaber dieses Routing-Codes sind.
- Wenn es eine ungerichtete Meldung und keine Frage ist und es Bedienstationen gibt, die die Unterdrückung aller nicht explizit bestellten ungerichteteten Meldungen gefordert haben (mit den Kommandos MODIFY-MSG-SUBSCRIPTION ..., DELIVER-OTHER-MSG=*NO oder ASR NOINF), dann werden diese Bedienstationen aus dem Empfängerkreis gestrichen.
- Wenn es eine ungerichtete Meldung ist und es Bedienstationen gibt, die eine Unterdrückung von Meldungen eines betreffenden Gewichts und Routing-Codes gefordert haben (Kommando ADD-CONSOLE-FILTER), dann werden diese Bedienstationen aus dem Empfängerkreis gestrichen.
- Wenn es eine ungerichtete Meldung ist und sie einen Meldungsschlüssel hat, und wenn es keine Frage oder eine sonstige, als „nicht bestellbar“ gekennzeichnete Meldung ist, und wenn es Meldungsbestellungen (mit dem MODIFY-MSG-SUBSCRIPTION-Kommando) für den betreffenden Meldungsschlüssel (oder Teile von ihm) gibt, dann werden die Bedienstationen, für die diese Meldungsbestellungen gelten, in den Empfängerkreis eingetragen.
- Wenn die Meldung einen Meldungsschlüssel hat und keine Frage ist, und wenn es Bedienstationen gibt, die die Unterdrückung von Meldungen mit diesem Meldungsschlüssel gefordert haben (Kommando SET-MSG-SUPPRESSION), dann werden diese Bedienstationen aus dem Empfängerkreis gestrichen.

Die höchste Priorität hat die Unterdrückung der Meldungszustellung anhand der Meldungsschlüssel. Eine Bedienstation kann also eine Gruppe von Meldungen mit dem Kommando MODIFY-MSG-SUBSCRIPTION bestellen (z.B. alle Meldungen mit der Meldungsklasse NBR) und aus dieser Menge einzelne Meldungsschlüssel (z.B. `NBR0740`) mit dem Kommando SET-MSG-SUPPRESSION ausklammern.

### NOINF-Funktion

Mit dem Kommando ASR kann eine Konsole in den NOINF-Zustand versetzt werden.
In diesem Zustand werden alle unbeantwortbaren, zu verteilenden Meldungen (durch „%“ gekennzeichnet und über einen Routing-Code zu senden) unterdrückt.

Dieselbe Wirkung erreicht man mit dem Kommando ADD-CONSOLE-FILTER, wenn für eine Konsole für alle Routing-Codes alle 5 Filterstufen beantragt werden, oder wenn das Kommando MODIFY-MSG-SUBSCRIPTION DELIVER-OTHER-MSG=*NO eingegeben wird.

### Nachrichtenunterdrückung durch Filterung

Der Operator hat die Möglichkeit, durch Einstellen von Filterstufen (mit dem Kommando ADD-CONSOLE-FILTER oder die Parameterservice-Anweisung SET-FILTER) die Ausgabe bestimmter Meldungsgruppen zu unterdrücken.

Es handelt sich dabei grundsätzlich um unbeantwortbare, zu verteilende Meldungen (durch „%“ gekennzeichnet und über einen Routing-Code zu senden), die über ein Gewicht verfügen, also ihren Ursprung in einer Meldungsdatei haben und über den Makro MSG7 oder MSG7X produziert wurden.

Es können 5 Filterstufen angegeben werden, die den „Weight Codes“ der Meldungen (Meldungsgewicht) folgendermaßen zugeordnet sind:

| Weight Code | Filterstufe | Bedeutung |
|---|---|---|
| bis 19 | 1 | Frei für Kunden |
| 20 - 39 | 2 | Rein informative Meldungen |
| 40 - 59 | 3 | Erhöht informative Meldungen |
| 60 - 79 | 4 | Wichtige informative Meldungen |
| 80 - 99 | 5 | Sehr wichtige informative Meldungen |

Z.B. werden bei eingeschalteter Filterstufe 2 alle Meldungen mit den Codes 20 bis 39 unterdrückt.

Die Einstellung der Filterstufen kann mit dem Kommando SHOW-CONSOLE-FILTER angezeigt werden. Mit dem Kommando ADD-CONSOLE-FILTER eingestellte Filterstufen können mit dem Kommando REMOVE-CONSOLE-FILTER wieder aufgehoben werden; mit der Parameterservice-Anweisung SET-FILTER eingestellte Filterstufen können nicht wieder zurückgenommen werden.

Welche Weight Codes den einzelnen Meldungen zugeordnet sind, ist in den Meldungsdateien beschrieben. Meldungsdateien können mit dem Dienstprogramm MSGMAKER angesehen werden. Die Systembetreuung kann außerdem mit MSGMAKER das Gewicht einer Meldung verändern.

### Meldungsbestellung anhand der Meldungsschlüssel

Für Bedienstationen ist es möglich, Meldungen anhand ihrer Meldungsschlüssel explizit anzufordern. Es werden dann u.U. nur diese bestellten Meldungen an der jeweiligen Bedienstation ausgegeben.

Die Meldungsbestellung erfolgt mit dem Kommando MODIFY-MSG-SUBSCRIPTION und der Angabe der vollständigen oder beliebig reduzierten Meldungsschlüssel.
Der Operand ADD-MSG-ID fügt Meldungen (zum bereits bestehenden Meldungsbestell-Umfang) hinzu, mit dem Operanden REMOVE-MSG-ID können zuvor bestellte Meldungen aus dem Meldungsbestell-Umfang wieder herausgenommen werden.

Ein weiterer Operand entscheidet, ob alle nicht explizit angeforderten Meldungen an dieser Bedienstation unterdrückt werden sollen. Mit DELIVER-OTHER-MSG=*NO werden nur die mit ADD-MSG-ID bestellten Meldungen sowie Fragen und explizit an diese Bedienstation gerichtete Meldungen zugestellt (das entspricht der ASR NOINF-Funktion).
Meldungsbestellungen haben keinen Einfluss auf die Zustellung der betreffenden Meldungen an anderen Bedienstationen.

Die Eingabe der Kommandos zur Meldungsbestellung ist an allen Typen von Bedienstationen zulässig, also sowohl an physikalischen Konsolen, logischen Konsolen ($CONSOLE-Anwendungen) und Dialogtasks mit dem Privileg OPERATING.
An physikalischen und logischen Konsolen beeinflussen die Kommandos die Meldungszustellung an der Bedienstation, von der aus sie eingegeben wurden.
In Dialogtasks mit dem Privileg OPERATING wirken die Kommandos auf einen System-Ereignisstrom (siehe auch Seite 608).

### Nachrichtenunterdrückung anhand der Meldungsschlüssel

Um gezielt die Ausgabe einzelner Meldungen auf eine Bedienstation oder ein berechtigtes Benutzerprogramm zu unterdrücken, steht dem Operator das Kommando SET-MSG-SUPPRESSION zur Verfügung. Die Meldungsausgabe von zu bis zu 1000 verschiedenen Meldungsnummern kann unterdrückt werden.

Die Meldung muss eine Nummer haben, sie muss also ihren Ursprung in einer Meldungsdatei haben und über den Makro MSG7 oder MSG7X produziert worden sein. Es können alle Typen von unbeantwortbaren Nachrichten (Kennzeichen %, + und !) unterdrückt werden.

Die Ausgabe von Meldungen einer angegebenen Meldungsnummer kann mit dem Kommando RESET-MSG-SUPPRESSION wieder zugelassen werden, die Vereinbarungen können mit SHOW-MSG-SUPPRESSION aufgelistet werden.
Diese Feinfilterung über die Angabe von Meldungsnummern ist an allen Bedienstationen bzw. berechtigten Benutzerpogrammen zulässig; eine Vereinbarung für andere Konsolen bzw. Benutzerprogramme kann allerdings nur von der Hauptbedienstation aus getroffen werden.

Durch die Vergabe entsprechender Werte für den Parameter OPR innerhalb der BS2000-Startup-Parameterdatei kann der Meldungsfluss bereits bei der Systemeinleitung reduziert werden (siehe Abschnitt „Konfiguration und Meldungsunterdrückung an Bedienstationen (OPR)“ auf Seite 104).

### 16.3.5 Nachrichtenaustausch der Operator

Die Operator haben die Möglichkeit, Meldungen über die Bedienstationen auszutauschen.

Von einer gegebenen Bedienstation aus lässt sich dabei eine Meldung an folgende Ziele senden:

- an eine andere Bedienstation (mit dem mnemotechnischen Gerätenamen).
- an Bedienstationen, denen ein bestimmtes Aufgabengebiet zugeordnet ist (mit dem Berechtigungsschlüssel, siehe Abschnitt „Aufgabengebiete und ihre Aufteilung auf Bedienstationen“ auf Seite 642).

Zwischen je zwei syntaktischen Einheiten dürfen Leerzeichen stehen. Beim Senden einer Meldung dürfen z.B. zwischen den syntakischen Einheiten „an“ und „-mid“ sowie „%“ Leerzeichen stehen. Siehe auch die jeweiligen Beispiele.

Folgende Formate werden beschrieben:

- Formate beim Senden von Operatormeldungen
- Formate beim Empfangen von Operatormeldungen
- Format beim Senden von Operatorantworten
- Format beim Empfangen von Operatorantworten

### Formate beim Senden von Operatornachrichten

```
⎧<rc ⎫          ⎧%⎫
⎨an  ⎬ [-mid]   ⎨ ⎬ text
⎩(mn)⎭          ⎩?⎭
```

< Spitze Klammer auf zur Einleitung des Berechtigungsschlüssels

rc Berechtigungsschlüssel (1 Zeichen)
Die Meldung geht an alle zugeordneten Bedienstationen und berechtigten Benutzerprogramme.
Der Schlüssel * bedeutet:
Die Meldung geht *in jedem Fall auch* an die Hauptbedienstation.

an Berechtigungsname eines berechtigten Benutzerprogramms (4 Zeichen).
Die Meldung geht an das berechtigte Benutzerprogramm, das mit diesem Berechtigungsnamen verknüpft ist.

mn Mnemotechnischer Gerätename einer Bedienstation (2 Zeichen)
Die Meldung geht an die bezeichnete Bedienstation.

\- Bindestrich (Minuszeichen)

mid Meldungsauftragskennzeichen (A..Z, 0-9, @, #, $);
Die letzten drei Zeichen werden als mid für die Ausgabe übernommen. Bei Angabe von weniger als drei Zeichen wird von links mit Nullen aufgefüllt.

% Prozentzeichen; Kennzeichen, dass die Meldung eine reine Mitteilung ist (keine Antwort erforderlich)

? Fragezeichen; Kennzeichen, dass die Meldung eine Frage ist, die vom Empfänger eine Antwort verlangt

text Beliebiger Text

## Formate beim Empfangen von Operatormeldungen

```
⎧%⎫ ⎧an  ⎫
⎨ ⎬ ⎨    ⎬ -mid.hhmmss text
⎩?⎭ ⎩(mn)⎭
```

% Prozentzeichen; Antwort nicht erforderlich

? Fragezeichen; der Absender verlangt eine Antwort

an Berechtigungsname des berechtigten Benutzerprogramms, von dem die Meldung stammt (4 Zeichen)

mn Mnemotechnischer Gerätename der Bedienstation, von der die Meldung stammt

\- Bindestrich

mid Meldungsauftragskennzeichen, Standardwert = 000 (A..Z, 0-9, @, #, $)

. Punkt als Trennzeichen

hhmmss
Tageszeit (6 Ziffern; hh = Stunde, mm = Minute, ss = Sekunde)

text Beliebiger Meldungstext

*Beispiele*

```
Eingabe von (K3): <A % THIS IS A MESSAGE WITH NO MID
Ausgabe an *)    : %(K3)-000.142423 THIS IS A MESSAGE WITH NO MID

Eingabe von (K4): <*-AKZ% THIS IS A MESSAGE WITH AN MID
Ausgabe an **)   : %(K4)-AKZ.142523 THIS IS A MESSAGE WITH AN MID

Eingabe von (K4): <*-123456789 % THIS IS A MESSAGE WITH A LONG MID
Ausgabe an **)   : %(K4)-789.142623 THIS IS A MESSAGE WITH A LONG MID

Eingabe von (K4): <*-@#$ % THIS IS A MESSAGE WITH SPECIAL CHARACTERS IN THE
MID
Ausgabe an **)   : %(K4)-@#$.142623 THIS IS A MESSAGE WITH SPECIAL CHARACTERS
IN THE MJID

Eingabe von (K4): <*-1 % THIS IS A MESSAGE WITH A SHORT MID
Ausgabe an **)   : %(K4)-001.142623 THIS IS A MESSAGE WITH A SHORT MID
```

*) Ausgabe erfolgt an alle Inhaber des Routing-Codes A

**) Ausgabe erfolgt an alle Inhaber des Routing-Codes *



Aus Kompatibiltätsgründen werden die letzten drei Zeichen des Meldungsauftragskennzeichens zur Ausgabe verwendet. Ein- und zweistelligen Meldungsauftragskennzeichen werden führende Nullen vorangestellt. Es sind auch Meldungsauftragskennzeichen mit mehr als acht Zeichen zulässig. Das erste Zeichen darf eine Ziffer oder eines der Zeichen @, $, # sein.

Diese Bedingungen für das Meldungsauftragskennzeichen entsprechen nicht den Vorschriften für das Kommandoauftragskennzeichen (siehe „Format bei der Eingabe von Operatorkommandos“ auf Seite 626).

### Format beim Senden von Operatorantworten

```
dest [-mid].[text]
```

dest    Ziel der Antwort/Kommandozusatzinformation
Dies ist der Absender der zu beantwortenden Meldung.

-    Bindestrich

mid    Meldungsauftragskennzeichen (A..Z, 0-9, @, #, $); führende Nullen dürfen entfallen (mid muss der Meldung, die beantwortet werden soll, entsprechen)

.    Punkt; Kennzeichen für Antwort auf Fragen

text    Beliebiger Antworttext
Bei Antworten auf Systemmeldungen sind nur bestimmte Texte zulässig.
Hilfe zu einzelnen Meldungen erhalten Sie mit /HELP-MSG-INFORMATION.

### Format beim Empfangen von Operatorantworten

```
  ⎧ rc ⎫
. ⎨    ⎬ [-mid].[text]
  ⎩(mn)⎭
```

| | |
|---|---|
| . | Punkt als Nachrichtentypkennzeichen einer Operator-Antwort |
| rc | Berechtigungsname des antwortenden berechtigten Benutzerprogramms (4 Zeichen) |
| mn | Mnemotechnischer Gerätename der antwortenden Bedienstation (2 Zeichen) |
| - | Bindestrich |
| mid | Meldungsauftragskennzeichen (A..Z, 0-9, @, #, $); wird bis zu 3 Zeichen mit führenden Nullen ergänzt. Standardwert = 000 |
| . | Punkt als Trennzeichen |
| text | Beliebiger Antworttext |

*Beispiel*

```
Fragesteller   Eingabe: (C2)-1? TO BE OR NOT TO BE ?
Frageempfänger Ausgabe: (C1)-001.092312 TO BE OR NOT TO BE ?
               Eingabe: (C1)-1. TO BE - THAT IS THE ANSWER
Fragesteller   Ausgabe: .(C2)-001. TO BE - THAT IS THE ANSWER
```

## 16.4 Verwendung mehrerer Bedienstationen

Die Bedienung des Systems kann auch von Benutzertasks betrieben werden, die das Privileg OPERATING besitzen.
Dieser Abschnitt beschreibt nur die Systembedienung von Konsolen (Bedienstationen).

BS2000/OSD bietet die Möglichkeit, mehrere Bedienstationen einzusetzen und ihnen die Aufgabengebiete so zuzuordnen, wie es der Betrieb eines Servers erfordert (siehe Abschnitt „Hauptbedienstation und Nebenbedienstationen“ auf Seite 639).

An S-Servern mit SKP 3970 können bis zu 40 Bedienstationen betrieben werden.

BS2000/OSD gestattet darüber hinaus dem Personal eines Data Centers, Datenstationen während des Systemlaufs als Bedienstationen zu verwenden; dazu kann das Software-Produkt OMNIS (mit OMNIS-PROP) verwendet oder es können Automatisierungs-Programme erstellt werden (siehe auch Abschnitt „Software-Produkte OMNIS, PROP-XT und OMNIS-PROP“ auf Seite 693).

Ein Schutzmechanismus (mit Kennwörtern) verhindert dabei den Mißbrauch dieser Funktion durch nichtprivilegierte Benutzer.

Der Einsatz dieser Funktion ist zum Beispiel dann angebracht, wenn die Systemverwaltung – ausnahmsweise – in das Operating eingreifen möchte, Teile des Operating automatisiert werden sollen oder gleichartige Aufgaben für mehrere Server von einem Operator bearbeitet werden sollen.

### Zuordnung von Aufgabengebieten zu Bedienstationen

Die Aufgabengebiete lassen sich auf die Bedienstationen folgendermaßen verteilen:

- Jedes Aufgabengebiet ist einer Bedienstation eindeutig zugeordnet: dies ist die strenge Trennung von Arbeiten nach Aufgabengebieten, wobei manchen Bedienstationen durchaus mehrere Aufgabengebiete zugeordnet sein dürfen.
- Einzelne Aufgabengebiete sind an mehreren Bedienstationen zu bearbeiten: dies ist eine unscharfe Aufteilung der Arbeiten nach Aufgabengebieten; sie ist nur in Sonderfällen sinnvoll.

Eine Übersicht über die Aufgabengebiete enthält der Abschnitt „Aufgabengebiete und ihre Aufteilung auf Bedienstationen“ auf Seite 642.

Die Zuweisung von Routing-Codes (Zuordnung zu Aufgabengebieten) erfolgt:

- bei logischen Konsolen mit generierten Berechtigungsnamen über den Parameterservice bzw. das ASR-Kommando
- bei logischen Konsolen mit dynamischen Berechtigungsnamen über Operator-Rollen

- bei physikalischen Konsolen über eine der beiden Möglichkeiten in Abhängigkeit vom Systemparameter NBCONOPI

### Auswirkungen der Zuordnungen im Systemlauf

Die Zuordnung von Aufgabengebieten zu Bedienstationen sind im Betrieb daran zu erkennen, dass Meldungen, die zu einem bestimmten Aufgabengebiet gehören, nur an der zugeordneten Bedienstation erscheinen (bzw. an den zugeordneten Bedienstationen); umgekehrt lassen sich Kommandos, die zu diesem Aufgabengebiet gehören, an der zugeordneten Bedienstation eingeben (bzw. an den zugeordneten Bedienstationen).
Ist der für die Eingabe eines bestimmten Kommandos notwendige Berechtigungsschlüssel keiner Bedienstation zugeordnet, so kann von der Bedienstation mit der Funktion „Hauptbedienstation“ dennoch dieses Kommando eingegeben werden, falls NBCONOPI=N eingestellt ist. Bei NBCONOPI=Y sind auch der Hauptbedienstation nur diejenigen Kommandos erlaubt, deren Berechtigung sie explizit besitzt. Soll ein Kommando in einem System generell nicht eingebbar sein, so ist es im Parameterservice mit dem Berechtigungsschlüssel $ zu schützen.

Wenn berechtigte Benutzerprogramme verwendet werden, so gelten die obenstehenden Aussagen für Bedienstationen und berechtigte Benutzerprogramme.

### Nachrichten zwischen den Bedienstationen

Die Operatoren haben die Möglichkeit, Nachrichten über die Bedienstationen auszutauschen.

Von einer gegebenen Bedienstation aus lässt sich dabei eine Nachricht an folgende Ziele senden:

- eine andere Bedienstation (mit dem mnemotechnischen Gerätenamen)
- an Bedienstationen, denen ein bestimmtes Aufgabengebiet zugeordnet ist (mit dem Berechtigungsschlüssel, siehe Abschnitt „Aufgabengebiete und ihre Aufteilung auf Bedienstationen“ auf Seite 642).

Die Formate der Nachrichten sind im Abschnitt „Nachrichtenformate“ auf Seite 623 zu finden.

| Kommando | Bedeutung |
|---|---|
| REQUEST-MAIN-CONSOLE-FUNCTION | Anfordern der Eigenschaft „Hauptbedienstation“ |
| SHOW-CONSOLE-STATUS | Anzeigen von Konfiguration und Status von Konsolen |

Tabelle 48: Kommandoübersicht zur Verwendung mehrerer Bedienstationen

### 16.4.1 Hauptbedienstation und Nebenbedienstationen

Unter den an einem Server angeschlossenen Bedienstationen hat eine für das System eine Sonderbedeutung: die Hauptbedienstation. Alle weiteren – sofern vorhanden – werden Nebenbedienstationen genannt. Hauptbedienstation ist zunächst immer diejenige, von der aus das System geladen wird (die server-abhängigen Geräteadressen der Bedienstationen sind im Handbuch „Systeminstallation“ [57] beschrieben). Erst nachdem das Laden des Systems abgeschlossen ist, sind auch die Nebenbedienstationen betriebsbereit.
Ab diesem Zeitpunkt gibt es zwischen den verfügbaren physikalischen Konsolen (= Bedienstationen) keine technischen Unterschiede mehr (mit Ausnahme der Teleservice-Anschlüsse). BS2000/OSD unterscheidet sie daher nur noch nach ihrer Funktion:

#### Hauptbedienstation

Diese Funktion ist zu jedem Zeitpunkt genau einer Bedienstation zugeordnet. Bei Ausfall dieser Bedienstation wird sie vom System automatisch einer anderen verfügbaren Bedienstation übertragen. Mit dem Kommando REQUEST-MAIN-CONSOLE-FUNCTIONS kann der Operator die Funktion auch gezielt einer anderen Bedienstation übergeben. Die jeweils als Hauptbedienstation wirksame Bedienstation hat folgende Eigenschaften:

- Sie erhält den Routing-Code * – auch dann, wenn er ihr nicht explizit zugewiesen wurde.
- Sie erhält alle so genannten „Emergency-Nachrichten“.
- Von ihr aus können alle Fragen und alle Anforderungen zu Kommando-Zusatzinformationen beantwortet werden, auch Emergency-Fragen (Ausnahme: an SCI-Kommandos gebundene und „;“-Fragen).

*bei NBCONOPI=N*

- Sie ist berechtigt, alle Kommandos auszuführen, deren Routing-Code keiner Konsole zugeordnet ist.
- Von dieser Bedienstation lässt sich die Zuordnung der Nebenbedienstationen sowie das Zu- und Wegschalten von Bedienstationen per Kommando regeln.
- Von dieser Bedienstation aus lässt sich die Zuordnung von Aufgabengebieten zu anderen Bedienstationen jederzeit ändern.
- Sie erhält in der Regel alle Meldungen, die sich keiner anderen Bedienstation zuordnen lassen (siehe auch Systemparameter MSGDEST im Anhang, Seite 749).

*zusätzlich bei NBCONOPI=Y*

- Von dieser Bedienstation lässt sich die Zuordnung von Aufgabengebieten zu logischen Konsolen mit generiertem Berechtigungsnamen mit dem ASR-Kommando ändern.
- Sie erhält alle Fragen, die an ihrem gewünschten Ausgabeort nicht ausgegeben werden konnten.

Wie bei allen Bedienstationen richten sich die Befugnisse der Hauptbedienstation danach, welche Aufgabengebiete ihr zugeordnet sind.



Bis zum Ende des Ladevorgangs fällt die Funktion Hauptbedienstation mit dem Gerät zusammen, von dem aus das System geladen wurde.
Die Funktion Hauptbedienstation lässt sich nur einer Bedienstation zuordnen, nicht aber einem berechtigten Benutzerprogramm. Lediglich das Recht zur Beantwortung offener Fragen kann mit dem Systemparameter NBREPLY an den Routing-Code * (Stern) geknüpft werden.

### Nebenbedienstation

Alle Bedienstationen, die nicht die Funktion Hauptbedienstation haben, werden vom System als Nebenbedienstation angesehen. Die Befugnisse einer Nebenbedienstation richten sich danach, welche Aufgabengebiete ihr zugeordnet sind.

### Umschalten der Funktion Hauptbedienstation

Das System schaltet die Funktion Hauptbedienstation in folgenden Situationen um:

- Auf Anforderung eines Operators.
  Von der Bedienstation aus, die die Funktion Hauptbedienstation erhalten soll, ist die Umschaltung anzufordern (mit dem Kommando REQUEST-MAIN-CONSOLE-FUNCTIONS). Das System verlangt daraufhin von der derzeitigen Hauptbedienstation eine Bestätigung. Trifft diese ein, wird die Umschaltung vorgenommen. Andernfalls weist das System die Anforderung zurück.
- Bei Ausfall der derzeitigen Hauptbedienstation.
  In diesem Fall schaltet das System automatisch auf eine anderen Bedienstation um:
  Für NBCONOPI=N wird die im Parameterservice vordefinierte Ersatzbedienstation verwendet (siehe Abschnitt „Anweisung DEFINE-CONSOLE“ auf Seite 109). Diese Auswahl kann vom Operator durch das CONSOLE-Kommando beeinflusst werden.
  Für NBCONOPI=Y wird die neue Hauptbedienstation durch einen internen Algorithmus ausgewählt.

### 16.4.2 Ersatzbedienstationen

Für jede Bedienstation kann bei NBCONOPI=N eine Ersatzbedienstation definiert werden. Fällt eine Bedienstation aus, wird die ihr zugeordnete Ersatzbedienstation automatisch aktiviert.
Werden Ersatzbedienstationen nicht im Parameterservice definiert, so werden sie vom System durch einen internen Algorithmus ausgewählt.
Der Operator hat die Möglichkeit, die Zuordnungen zu verändern oder auf Ersatzbedienstationen umzuschalten (siehe Kommando CONSOLE).

Bei NBCONOPI=Y gibt es die Funktion „Ersatzbedienstation“ nicht.

### 16.4.3 Remote-Service / Teleservice

Der Service führt die vertraglich vereinbarten Service-Tätigkeiten in der Regel aus der Ferne als Remote-Service durch. Die Verbindung zur Service-Zentrale erfolgt entweder über das angeschlossene Modem (Teleservice) oder über eine Internetverbindung (AIS Connect).

Remote-Service ermöglicht die Fernbedienung, Ferndiagnose und Fernwartung von Datenverarbeitungs-Systemen. Über Fernleitung werden auftretende Probleme von Spezialisten in Service-Zentren untersucht. Der Zugriff auf DV-Systeme ist bei laufendem Betriebssystem und ohne Betriebssystem möglich.

Remote-Service wird vom Service betrieben und deshalb in diesem Handbuch nicht näher beschrieben.

## 16.4.4 Aufgabengebiete und ihre Aufteilung auf Bedienstationen

### Aufgabengebiete bei der Bedienung von BS2000/OSD

Eine Übersicht über die Aufgabengebiete bei der Bedienung von BS2000/OSD gibt die nachfolgende Tabelle. In der Tabelle sind auch – in alphabetischer Reihenfolge – die Kommandos angegeben, die zu den einzelnen Aufgabengebieten gehören. Diese Aufteilung der Kommandos gilt standardmäßig; sie kann bei der Systemeinleitung geändert werden (für jedes Kommando gesondert, siehe Abschnitt „Anweisung SET-CMD-CODE“ auf Seite 110).

Für einige Kommandos ist die standardmäßige Zuordnung zu einem Aufgabengebiet und damit auch die Zuordnung eines Berechtigungsschlüssels vom Systemparameter NBCONOPI abhängig. Einige Kommandos, die bei NBCONOPI=N standardmäßig mit dem Berechtigungsschlüssel $ geschützt sind, erhalten bei eingeschaltetem Operator-Logon (NBCONOPI=Y) einen anderen Berechtigungsschlüssel. Sie werden in der Tabelle gesondert ausgewiesen.

Jedes Aufgabengebiet ist mit einem Berechtigungsschlüssel (BS), auch Routing-Code (RC) genannt, versehen, der bei der Zuordnung jeweils anzugeben ist.
Die Bedeutung dieser Berechtigungsschlüssel ist ab Seite 649 beschrieben.
Über die Zuordnung von Aufgabengebieten zu Bedienstationen gibt der Abschnitt „Zuordnung von Aufgabengebieten zu Bedienstationen“ auf Seite 652 Auskunft.

Eine Zuordnung „Operatorkommandos zu Berechtigungsschlüssel“ finden Sie im Handbuch „Kommandos“ [27].



ISP-Kommandos können aus Kompatibilitätsgründen noch angegeben werden. Ausführlich beschrieben sind diese Kommandos nur noch im Handbuch „Operator-Kommandos (ISP-Format)“ [37].

Die Tabelle gilt i.A. nur bei NBCONOPI=N. Ausnahmen sind gesondert ausgezeichnet.

| BS / RC | Aufgabengebiet | Standard für Kommando |
|---|---|---|
| A | Systemverwaltung | CHANGE-SERSLOG-FILE,<br>SHOW-SERSLOG-STATUS,<br>START-/STOP-SERSLOG |
| C | Bedienung des Datenkommunikationssystems | alle BCAM-Kommandos, z.B.:<br>DCSTART, BCEND |
| | Bedienung und Steuerung des File Transfer | openFT-Kommandos, siehe Handbuch „openFT“ [23], z.B.<br>ADD-FT-PARTNER,<br>START-/STOP-FT |

Tabelle 49: Berechtigungsschlüssel (BS / RC) - Aufgabengebiete - Kommandos (Teil 1 von 7)

| BS / RC | Aufgabengebiet | Standard für Kommando |
|---|---|---|
| D | Bedienung von Plattenspeichern | CHANGE-DISK-MOUNT,<br>CHECK-DISK-MOUNT,<br>DRV-Kommandos, siehe Handbuch „DRV“ [17],<br>z.B.<br>SHOW-DRV-STATUS,<br>START-/STOP-DRV-DUAL-MODE,<br>SET-DISK-DEFAULTS,<br>SET-DISK-PARAMETER,<br>UNLOCK-DISK |
| E | Allgemeine Aufgaben und Befugnisse<br><br>Bei **NBCONOPI=Y** bekommen einige Kommandos, die sonst den BS $ haben, den BS E zugewiesen. | CANCEL-RUN-PROCESS,<br>INFORM-ALL-JOBS,<br>INFORM-JOB,<br>MODIFY-MSG-FILE-ASSIGNMENT,<br>MODIFY-MSG-SUBSCRIPTION,<br>PROTECT-FITC-APPLICATION,<br>REQUEST-MAIN-CONSOLE-FUNCTIONS,<br>SHOW-CJC-STATUS,<br>SHOW-DEVICE-CONFIGURATION,<br>SHOW-DEVICE-DEPOT,<br>SHOW-DEVICE-STATUS,<br>SHOW-DISK-DEFAULTS,<br>SHOW-DISK-STATUS,<br>SHOW-JOB-STATUS,<br>SHOW-MASTER-CATALOG-ENTRY,<br>SHOW-MOUNT-PARAMETER,<br>SHOW-PUBSET-ATTRIBUTES,<br>SHOW-RESOURCE-ALLOCATION,<br>SHOW-RESOURCE-REQUESTS,<br>SHOW-SHARED-PUBSET,<br>SHOW-SYSTEM-STATUS,<br>SHOW-TAPE-STATUS,<br>SHOW-USER-STATUS<br>SHOW-XCS-PUBSET,<br>START-/STOP-DIALOG-APPLICATION |

Tabelle 49: Berechtigungsschlüssel (BS / RC) - Aufgabengebiete - Kommandos (Teil 2 von 7)

| BS / RC | Aufgabengebiet | Standard für Kommando |
|---|---|---|
| G | Geräteverwaltung | ADD-IO-UNIT,<br>ATTACH-DEVICE,<br>DETACH-DEVICE,<br>INCLUDE-DEVICE-CONNECTION,<br>MODIFY-IO-UNIT,<br>MODIFY-MOUNT-PARAMETER,<br>MOUNT-NET-STORAGE,<br>REMOVE-DEVICE-CONNECTION,<br>REMOVE-IO-UNIT,<br>SHOW-NET-STORAGE,<br>START-/STOP-CONFIGURATION-UPDATE,<br>UMOUNT-NET-STORAGE,<br>UNLOCK-DEVICE |
| H | Hardware-Wartung | |
| J | Steuerung und Kontrolle von Jobs<br><br>Bei **NBCONOPI=Y** bekommen einige Kommandos, die sonst den BS $ haben, den BS J zugewiesen. | HOLD-JOB,<br>HOLD-JOB-CLASS,<br>HOLD-JOB-STREAM,<br>HOLD-TASK,<br>MODIFY-JOB,<br>MODIFY-JOB-CLASS,<br>MODIFY-JOB-STREAM,<br>MODIFY-JV,<br>MODIFY-RESOURCE-COLLECTION,<br>RESUME-JOB,<br>RESUME-JOB-CLASS,<br>RESUME-JOB-STREAM,<br>RESUME-TASK,<br>SHOW-JV,<br>START-/STOP-JOB-STREAM,<br>START-/STOP-RESOURCE-COLLECTION |
| K | OPR-Kommandoverwaltung | CONNECT-CMD-SERVER |
| N | Überwachung von Remote-Spool | REDIRECT-PRINT-JOB |
| O | Bedienung von Schnelldruckern | HOLD-PRINT-JOB (Alias HOLD-SPOOLOUT),<br>RESUME-PRINT-JOB (Alias RESUME-SPOOLOUT) |
| P | Steuerung von Tasks | CANCEL-JOB,<br>CHANGE-TASK-CPU-LIMIT,<br>CHANGE-TASK-PRIORITY,<br>ENTER-JOB,<br>FORCE-JOB-CANCEL,<br>INFORM-PROGRAM,<br>SEND-MSG |

Tabelle 49: Berechtigungsschlüssel (BS / RC) - Aufgabengebiete - Kommandos (Teil 3 von 7)

| BS / RC | Aufgabengebiet | Standard für Kommando |
|---|---|---|
| R | Überwachung und Steuerung des Systemverhaltens | ACTIVATE-SNAPSHOT,<br>CANCEL-PUBSET-EXPORT/-IMPORT,<br>CHANGE-CONSLOG-FILE,<br>CREATE-/DELETE-PAGING-FILE,<br>DEACTIVATE-SNAPSHOT,<br>EXPORT-PUBSET,<br>EXTEND-PAGING-AREA,<br>FORCE-PUBSET-EXPORT,<br>HOLD-SUBSYSTEM,<br>IMPORT-PUBSET,<br>MODIFY-MEMORY-PARAMETERS,<br>MODIFY-MSCF-CONNECTION,<br>MODIFY-MSCF-ENVIRONMENT,<br>MODIFY-NSM-ENVIRONMENT,<br>MODIFY-PAGING-AREA-ATTRIBUTES,<br>MODIFY-PCS-OPTION,<br>MODIFY-SYSTEM-BIAS,<br>MODIFY-TASK-CATEGORIES,<br>REDUCE-PAGING-AREA,<br>RELEASE-CLUSTER-RECOVERY-LOCK,<br>RESUME-SUBSYSTEM,<br>SET-DSSM-OPTIONS,<br>SET-MSCF-ENVIRONMENT,<br>SET-RESTART-OPTIONS,<br>SET-XCS-PUBSET,<br>SHOW-MEMORY-CONFIGURATION,<br>SHOW-PAGING-CONFIGURATION,<br>SHOW-PCS-OPTION,<br>SHOW-RESTART-OPTIONS,<br>SHOW-SNAPSHOT-STATUS,<br>SHOW-SUBSYSTEM-STATUS,<br>SHOW-TRACE-STATUS,<br>SHUTDOWN,<br>START-/STOP-MSCF-CONNECTION,<br>START-/STOP-SUBSYSTEM,<br>START-/STOP-TRACE |
| S | Steuerung des SPOOLOUT-Betriebs<br>SPOOL laden | MODIFY-PRINTER-OUTPUT-STATUS,<br>MODIFY-TAPE-OUTPUT-STATUS,<br>SHOW-ACTIVE-SPOOL-DEVICES,<br>SHOW-PRINT-JOB-STATUS,<br>START-/STOP-PRINTER-OUTPUT,<br>START-/STOP-TAPE-OUTPUT,<br>START-/STOP-TAPE-REPLAY |

Tabelle 49: Berechtigungsschlüssel (BS / RC) - Aufgabengebiete - Kommandos (Teil 4 von 7)

| BS / RC | Aufgabengebiet | Standard für Kommando |
|---|---|---|
| T | Bedienung von Magnetbandgeräten | ADD-DEVICE-DEPOT,<br>CHANGE-TAPE-MOUNT,<br>CHECK-TAPE-MOUNT,<br>REMOVE-DEVICE-DEPOT,<br>UNLOAD-TAPE |
| U | Dateiverwaltung | |
| W,X,Y,Z | zur freien Verfügung | |
| B,F,I,L,M,<br>Q,V, 0..8 | reserviert | |
| 9 | VM2000-Verwaltung | alle VM2000-Kommandos |
| # | POSIX | |
| @ | Sonderbedeutung<br>(keine Berechtigung nötig) | ADD-CONSOLE-FILTER,<br>DISCONNECT-CMD-SERVER,<br>GRANT-PROP-CONNECTION,<br>HELP-MSG-INFORMATION,<br>MODIFY-CONSOLE-OPTIONS,<br>MODIFY-USER-PROTECTION,<br>RELEASE-OPERATOR-ROLE,<br>REMOVE-CONSOLE-FILTER,<br>REQUEST-OPERATOR-ROLE,<br>RESET-MSG-BUFFER,<br>RESET-MSG-SUPPRESSION,<br>SET-MSG-SUPPRESSION,<br>SHOW-CMD-ATTRIBUTES,<br>SHOW-CONSLOG-ATTRIBUTES,<br>SHOW-CONSOLE-FILTER,<br>SHOW-CONSOLE-OPTIONS,<br>SHOW-CONSOLE-STATUS,<br>SHOW-FILE-ATTRIBUTES,<br>SHOW-GS-COMPLEX-CONFIGURATION,<br>SHOW-GS-STATUS,<br>SHOW-GS-VOLUME-ATTRIBUTES,<br>SHOW-IOCF,<br>SHOW-MSCF-CONFIGURATION,<br>SHOW-MSG-SUBSCRIPTION,<br>SHOW-MSG-SUPPRESSION,<br>SHOW-NSM-CONFIGURATION,<br>SHOW-OPERATOR-ATTRIBUTES,<br>SHOW-OPERATOR-ROLE,<br>SHOW-PENDING-MSG, |

Tabelle 49: Berechtigungsschlüssel (BS / RC) - Aufgabengebiete - Kommandos (Teil 5 von 7)

| BS / RC | Aufgabengebiet | Standard für Kommando |
|---|---|---|
| @<br>(Forts.) | Sonderbedeutung<br>(keine Berechtigung nötig) | SHOW-PUBSET-CACHE-ATTRIBUTES,<br>SHOW-PUBSET-DEFINITION-FILE,<br>SHOW-PUBSET-IMPORT-EXPORT,<br>SHOW-PUBSET-OCCUPATION<br>SHOW-PUBSET-PROCESSING<br>SHOW-PUBSET-RESTRICTION<br>SHOW-PUBSET-SPACE-ALLOCATION,<br>SHOW-PUBSET-SPACE-DEFAULTS,<br>SHOW-SDF-PARAMETERS,<br>SHOW-SPACE-SATURATION-LEVELS<br>SHOW-SYSTEM-INFORMATION,<br>SHOW-SYSTEM-PARAMETERS,<br>SHOW-XCS-OCCUPATION |
| | zusätzlich bei **NBCONOPI=Y** | EXIT-JOB,<br>SET-LOGON-PARAMETERS |
| * | Hauptbedienstation | CONSOLE<br>(bei NBCONOPI=Y ist dieses Kommando nicht mehr erlaubt)<br>MODIFY-SDF-PARAMETERS |
| $ | Sonderbedeutung<br>(Kommando ist generell gesperrt) | ADD-MASTER-CATALOG-ENTRY,<br>ADD-USER,<br>COMPARE-DISK-FILES<br>LOCK-USER,<br>MODIFY-MASTER-CATALOG-ENTRY,<br>MODIFY-SPACE-SATURATION-LEVELS,<br>MODIFY-USER-ATTRIBUTES,<br>MODIFY-USER-PUBSET-ATTRIBUTES,<br>PRINT-DOCUMENT (PRINT-FILE),<br>REMOVE-MASTER-CATALOG-ENTRY,<br>REMOVE-USER,<br>SHOW-FILE-TRANSFER,<br>SHOW-PRINTER-POOLS,<br>SHOW-SPOOL-CHARACTER-SETS<br>SHOW-SPOOL-DEVICES<br>SHOW-SPOOL-FORMS<br>SHOW-SPOOL-PARAMETERS<br>SHOW-USER-ATTRIBUTES,<br>UNLOCK-USER<br>WRITE-SPOOL-TAPE |

Tabelle 49: Berechtigungsschlüssel (BS / RC) - Aufgabengebiete - Kommandos (Teil 6 von 7)

| BS / RC | Aufgabengebiet | Standard für Kommando |
|---|---|---|
| $ (Forts.) | bei **NBCONOPI=Y** ändert sich der BS von $ nach E | ADD-/SET-FILE-LINK,<br>ADD-PASSWORD,<br>COPY-FILE,<br>CREATE-FILE,<br>CREATE-FILE-GENERATION,<br>CREATE-FILE-GROUP,<br>CREATE-TAPE-SET,<br>DELETE-FILE,<br>DELETE-FILE-GENERATION,<br>DELETE-FILE-GROUP,<br>DELETE-SYSTEM-FILE,<br>DELETE-TAPE-SET,<br>EXPORT-FILE,<br>EXTEND-TAPE-SET,<br>IMPORT-FILE,<br>MODIFY-FILE-ATTRIBUTES,<br>MODIFY-FILE-GENERATION-SUPPORT,<br>MODIFY-FILE-GROUP-ATTRIBUTES,<br>REMOVE-PASSWORD |
| | bei **NBCONOPI=Y** ändert sich der BS von $ nach J | CREATE-JV, DELETE-JV,<br>MODIFY-JV-ATTRIBUTES,<br>REMOVE-JV-LINK,<br>SET-JV-LINK,<br>SHOW-JV-ATTRIBUTES,<br>SHOW-JV-LINK |

Tabelle 49: Berechtigungsschlüssel (BS / RC) - Aufgabengebiete - Kommandos (Teil 7 von 7)

### Beschreibung der einzelnen Aufgabengebiete

A Systemverwaltung

Dieses Aufgabengebiet umfasst das Aktivieren bzw. Deaktivieren des Software-Error-Logging und das Wechseln der SERSLOG-Datei.

C Bedienung des Datenkommunikationssystems

Dieses Aufgabengebiet umfasst z.B. folgende Tätigkeiten:
- Starten des Datenkommunikationssystems
- Definieren und Aktivieren von Kommunikationspartnern
- Aktivieren von Leitungen
- Definieren und Aktivieren von Routen zu Kommunikationspartnern

Jede zugeordnete Bedienstation erhält darüber hinaus Meldungen, die auf Fehler im Datenkommunikationssystem hinweisen.

D Bedienung von Plattenspeichern

Dieses Aufgabengebiet umfasst das Bereitstellen von Datenträgern (Platten).
Jede zugeordnete Bedienstation erhält darüber hinaus Meldungen, die auf Unstimmigkeiten bei den Archivnummern oder auf andere Fehler hinweisen.

E Allgemeine Aufgaben und Befugnisse

Dieses Aufgabengebiet umfasst keine genau festgelegten Tätigkeiten; vielmehr sollten die dazugehörigen Kommandos jeder Bedienstation zur Verfügung stehen.

G Geräteverwaltung

Dieses Aufgabengebiet umfasst z.B. folgende Tätigkeiten:
- Wegschalten eines Gerätes vom System
- Zuschalten eines Gerätes zum System
- Dynamische I/O-Konfigurationsänderung
- Net-Storage verbinden und trennen

Jede zugeordnete Bedienstation erhält darüber hinaus Meldungen, die auf Unstimmigkeiten hinweisen.

H Hardware-Wartung

Jede zugeordnete Bedienstation erhält Meldungen, die auf Fehler im Server hinweisen und die vom Wartungspersonal ausgewertet werden.

J Steuerung und Kontrolle von Jobs

Jede zugeordnete Bedienstation erhält Meldungen, die den Beginn, Besonderheiten oder Fehler bei der Bearbeitung von Jobs anzeigen, so dass der Operator den Ablauf von Jobs verfolgen kann.

K OPR-Kommandoverwaltung

Berechtigte Benutzerprogramme können Operatorkommandos definieren. Sie können die so definierten Kommandos auch wieder löschen, und sie können sich für die Bearbeitung bereits eingetragener Operatorkommandos zuständig erklären und die Zuständigkeit auch wieder kündigen.

N Überwachung von Remote-Spool

O Bedienung von Schnelldruckern

Dieses Aufgabengebiet umfasst z.B. folgende Tätigkeiten:
- Veranlassen eines Probedrucks
- Montieren eines Dias im Laserdrucker

Jede zugeordnete Bedienstation erhält darüber hinaus Meldungen, die auf Fehler bei Schnelldruckern hinweisen.

P Steuerung von Tasks

Dieses Aufgabengebiet umfasst z.B. folgende Tätigkeiten:
- Festlegen von Prioritäten für einzelne Jobs
- Starten von Batch-Jobs
- Abbrechen von Benutzer-Jobs

R Überwachung und Steuerung des Systemverhaltens

Dieses Aufgabengebiet umfasst z.B. folgende Tätigkeiten:
- Ändern der kategorie-spezifischen Vorgaben zu Auftrags-Anzahl und I/O-Priorität
- Beenden des Systemlaufs

Jede zugeordnete Bedienstation erhält darüber hinaus Meldungen, die Systemfehler oder Server-Fehler anzeigen oder die anzeigen, dass ein Benutzer mehr Speicherplatz auf gemeinschaftlichen Datenträgern angefordert hat, als ihm zusteht.

S Steuerung des SPOOLOUT-Betriebs

Dieses Aufgabengebiet umfasst z.B. die Zuordnung von Ausgabegeräten zu SPOOLOUT-Jobs. Jede zugeordnete Bedienstation erhält darüber hinaus Meldungen, die auf Unstimmigkeiten hinweisen.

T Bedienung von Magnetbandgeräten

Dieses Aufgabengebiet umfasst das Bereitstellen bzw. das Wechseln von Datenträgern. Jede zugeordnete Bedienstation erhält darüber hinaus Meldungen, die auf Fehler hinweisen.

U Dateiverwaltung

W,X,Y,Z
Zur freien Verfügung. Diese Aufgabengebiete stehen dem Benutzer für Sonderzwecke frei zur Verfügung (z.B. für Spezialkommandos).

B,F,I,L,M,Q,V,1,2,3,4,5,6,7,8
Reserviert für künftige Systemerweiterungen.
Diese Aufgabengebiete sind derzeit nicht definiert, können aber in zukünftigen Versionen definiert werden.

9 Reserviert für VM2000-Verwaltung

# POSIX

*Berechtigungsschlüssel mit Sonderbedeutungen*

0 Meldungen mit diesem Berechtigungsschlüssel sind nicht bestellbar. Meldungen ohne Antwort werden nur in der CONSLOG-Datei hinterlegt. Meldungen mit Antwort (Fragen) werden der Hauptbedienstation zugestellt.

* Hauptbedienstation
Dieses Aufgabengebiet ist immer auch der aktuellen Hauptbedienstation zugeordnet. Ob ein Kommando mit diesem Berechtigungsschlüssel tatsächlich von der Hauptbedienstation eingegeben wurde, wird nur beim Kommando CONSOLE geprüft.

Darüber hinaus erhält jede zugeordnete Bedienstation Meldungen, die über den Zustand von Abrechnungsdateien Auskunft geben.

@ Meldungen mit diesem Berechtigungsschlüssel sind bestellbar (Kommando MODIFY-MSG-SUBSCRIPTION). Meldungen ohne Antwort werden nur in der CONSLOG-Datei hinterlegt. Meldungen mit Antwort (Fragen) werden der Hauptbedienstation zugestellt. Kommandos mit diesem Berechtigungsschlüssel sind ungeschützt; jeder Operator hat die Berechtigung, sie einzugeben.
Das Zuweisen des Schlüssels @ an Bedienstationen hat daher praktisch keinen Sinn. Sinnvoll hingegen ist die Zuordnung des Schlüssels an Meldungen oder Kommandos.

$ Ein mit dem Schlüssel $ geschütztes Kommando ist grundsätzlich keinem Benutzer erlaubt. Der Berechtigungsschlüssel $ kann zwar beliebigen Bedienstationen und berechtigten Benutzerprogrammen zugeordnet werden, aber er berechtigt nicht zur Eingabe eines mit $ geschützten Kommandos. Als Verteilerschlüssel gibt es keine Sonderbedeutung für den Schlüssel $.

### Zuordnung von Aufgabengebieten zu Bedienstationen

Zum Zuordnen von Aufgabengebieten zu Bedienstationen gibt es folgende Möglichkeiten:

*bei NBCONOPI=N*

- Beim Systemstart

  Über die Startup-Parameterdatei können während der Startup-Phase Bedienstationen (nicht berechtigten Benutzerprogrammen) mit der SET-CODE-Anweisung (SET-CODE CODE=rc,CONSOLE=mn) Aufgabengebiete zugeordnet werden (siehe Anweisung SET-CODE, Seite 111).

- Während des Systemlaufs

  Von der jeweiligen Hauptbedienstation aus lässt sich mit dem Kommando ASR jede gewünschte Zuordnung für physikalische Bedienstationen und berechtigte Benutzerprogramme mit generierten Berechtigungsnamen herstellen.
  In Abhängigkeit von den Systemparametern ASRSW1 und ASRSW2 kann man von einer Bedienstation oder von einem berechtigten Benutzerprogramm mit generiertem Berechtigungsnamen (sofern dort das Kommando ASR zugelassen ist) mit dem Kommando ASR jede gewünschte Zuordnung für diese Bedienstationen oder für dieses berechtigte Benutzerprogramm einstellen.
  ASR-Kommandos zur Veränderung der Zuteilung von Aufgabengebieten werden abgewiesen, wenn sie von berechtigten Benutzerprogrammen mit dynamischen Berechtigungsnamen oder in Bezug auf solche eingegeben werden. Berechtigte Benutzerprogramme mit dynamischen Berechtigungsnamen ordnen sich ihre Aufgabengebiete mit dem Kommando REQUEST-OPERATOR-ROLE selbst zu.

  Die Funktion „Hauptbedienstation“ lässt sich nur dann umschalten, wenn die bisherige Hauptbedienstation zustimmt.

  Fällt die jeweilige Hauptbedienstation aus, so schaltet das System die Funktion „Hauptbedienstation“ automatisch auf eine andere Bedienstation um; dabei wird die im Parameterservice vordefinierte Ersatzbedienstation verwendet (siehe Abschnitt „Anweisung DEFINE-CONSOLE“ auf Seite 109). Diese Auswahl kann vom Operator durch das CONSOLE-Kommando beeinflusst werden.

*bei NBCONOPI=Y*

Die in der Startup-Parameterdatei mit der Anweisung SET-CODE der IPL-Konsole zugewiesenen Routing-Codes werden als „implizite Routing-Codes“ bezeichnet.
Ihre Wirkung wird über den Systemparameter NBIMPRCA gesteuert.
Bei NBIMPRCA=Y wirken sie während des gesamten Systemlaufs, bei NBIMPRCA=N nur bis „System Ready“ und ab Systembeendigung (nach erfolgtem Shutdown).

„Implizite Routing-Codes“ wirken grundsätzlich nur als Routing-Codes (zur Verteilung von Meldungen) und nicht als Berechtigungsschlüssel (zur Eingabe von Kommandos).

– Beim Systemstart und nach Systembeendigung

  Die Hauptbedienstation erhält alle Meldungen der Routing-Codes, die als „implizite Routing-Codes“ festgelegt wurden.

– Während des Systemlaufs (bis Shutdown)

  Bei NBIMPRCA=Y erhält die Hauptbedienstation alle Meldungen der „impliziten Routing-Codes“, sofern diese keiner anderen Bedienstation explizit zugeordnet sind.

  Eine Bedienstation kann sein:
  – operabel / inoperabel
  – aktiv / inaktiv
  – berechtigt / unberechtigt

  Eine Bedienstation ist inoperabel, wenn sie systemintern als „defekt“ verwaltet wird, was auch für nicht-existierende Geräte oder Geräte im Zustand DETACHED gilt. Alle anderen Bedienstationen sind operabel.
  Eine operable Bedienstation wird durch das Kommando SET-LOGON-PARAMETERS aktiv und nach dem Kommando EXIT-JOB wieder inaktiv.
  Eine aktive Bedienstation erhält durch das Kommando REQUEST-OPERATOR-ROLE verschiedene Berechtigungsschlüssel, die ihre Berechtigungsmenge darstellen.
  Zur Eingabe von @-Kommandos sind aktive Bedienstationen immer berechtigt.
  Inaktiven Bedienstationen stehen nur folgende Kommandos zur Verfügung:
  SET-LOGON-PARAMETERS und SHOW-PENDING-MSG.

  Die erste Bedienstation, an der das Kommando SET-LOGON-PARAMETERS erfolgreich eingegeben wurde, erhält die Eigenschaft „Hauptbedienstation“. Sie ist nun aktiv, hat aber nur die Berechtigung zur Eingabe von @-Kommandos.

  Die Funktion „Hauptbedienstation“ lässt sich nur dann umschalten, wenn die bisherige Hauptbedienstation zustimmt.

  Fällt die jeweilige Hauptbedienstation aus, so schaltet das System die Funktion „Hauptbedienstation“ automatisch auf eine andere Bedienstation um; dabei wird zunächst in der Menge der aktiven und danach in der Menge der operablen Bedienstationen gesucht. Auch ein EXIT-JOB-Kommando von der Hauptbedienstation leitet diese Suche ein.

Mit ASR-Berechtigung kann die Hauptbedienstation jede gewünschte Zuordnung von Routing-Codes an berechtigte Benutzerprogramme mit generiertem Berechtigungsnamen herstellen.

*Hinweis*

Das Zuordnen einer Bedienstation zum Aufgabengebiet @ hat praktisch keine Wirkung.

Es ist auch möglich, einen Berechtigungsschlüssel (z.B. A, C, R, U usw.) überhaupt keiner Bedienstation zuzuordnen. Wenn Berechtigungsschlüssel keiner Bedienstation zugeordnet sind, geschieht Folgendes:

- Meldungen, die keine Antwort erfordern, werden auf keiner Bedienstation ausgegeben. Sie werden nur in die Protokolldatei geschrieben (beim Schlüssel @ ist dies immer der Fall).
- Meldungen, die eine Antwort des Operators erfordern, werden an der Hauptbedienstation ausgegeben. Auch über den Schlüssel @ verteilte Meldungen werden an der Hauptbedienstation ausgegeben, wenn sie eine Antwort des Operators erfordern.
- Kommandos mit Berechtigungsschlüsseln, die keiner Bedienstation zugeordnet sind, werden bei NBCONOPI=N vom System akzeptiert, wenn sie von der Hauptbedienstation eingegeben werden.
  Ausnahme: Kommandos mit dem Schlüssel @ werden immer akzeptiert.

# 16.5 Rückmeldungen

## 16.5.1 Kommandobeendigungsmeldungen

Kommandobeendigungsmeldungen erkennt man am Meldungstyp „!“ als erstes Zeichen der Ausgabe. Folgende Kommandobeendigungsmeldungen sind möglich:

NBR0740 `COMMAND COMPLETED 'ASTOP' ; RESULT: (SC2=00, SC1=00, MC=CMD0001); DATE:<date>`
erscheint, wenn das Kommando fehlerfrei ausgeführt wurde.

NBR0741 `COMMAND SUCCESS UNCERTAIN 'SHOW-CONSLOG'; RESULT: (SC2=02, SC1=00, MC=NBR0741)`
erscheint, wenn während der Kommandobearbeitung ein Fehler auftrat, der die Terminierung der Operatortask zur Folge hatte.

NBR0741 `COMMAND SUCCESS UNCERTAIN 'SPECIAL'; RESULT: (SC2=02, SC1=64, MC=NBR0741)`
erscheint, wenn während der Kommandobearbeitung in einem berechtigten Benutzerprogramm die Verbindung zur DCAM-Anwendung $CONSOLE abgebaut wurde.

NBR0742 `COMMAND NOT ALLOWED 'ASR'; RESULT: (SC2=00, SC1=64, MC=CMD0216)`
erscheint, wenn die Bedienstation, von der aus das Kommando ausgegeben wurde, nicht berechtigt ist, dieses Kommando auszugeben.

NBR0743 `COMMAND NOT AVAILABLE 'SPECIAL'; RESULT: (SC2=00, SC1=130, MC=CMD0200)`
erscheint, wenn die Routine, die das Kommando verarbeitet, nicht im System ist, oder wenn zwischen der DCAM-Anwendung $CONSOLE und dem berechtigten Benutzerprogramm, das das Kommando ausführen soll, keine Verbindung besteht.
Beispiel: `/.FSTAT DADM xxx`
Falls TDADM nicht geladen ist, wird dieses Kommando mit der Meldung `NBR0743` zurückgewiesen.

NBR0744 `COMMAND NOT FOUND 'ADR'; RESULT: (SC2=00, SC1=64, MC=NBR0744)`
erscheint, wenn ein Kommando eingegeben wurde, das kein Operator-Kommando ist. Beispiel: `/.A1234567 ADR HELP`

NBR0745 `COMMAND ID INVALID '.1WRONGID'; RESULT: (SC2=00, SC1=01, MC=NBR0745)`
erscheint, wenn ein Kommandoauftragskennzeichen ein ungültiges Zeichen enthält. Beispiel: `/.1WRONGID ASR HELP`

NBR0746 `COMMAND CANCELLED 'RUN'; RESULT: (SC2=02, SC1=00, MC=NBR0746)`
erscheint, wenn ein RUN-Kommando mit dem Kommando CANCEL-RUN-PROCESS zurückgenommen wurde oder wenn wegen Abbruch der Verbindung zur Bedienstation die zugehörige Operator-Task terminiert wurde, bevor das Kommando von ihr bearbeitet werden konnte.

NBR0747 `COMMAND ID AMBIGUOUS 'ABC'; RESULT: (SC2=00, SC1=64, MC=NBR0747)`
erscheint, wenn ein Kommando mit einem nicht eindeutigen Kommandoauftragskennzeichen eingegeben wurde.
Das Kommandoauftragskennzeichen muss nur bei Kommandos eindeutig sein, die eine Kommandozusatzinformation verlangen können und deren Bearbeitung in einem berechtigten Benutzerprogramm erfolgt.

NBR0748 `SYSTEM SHUTDOWN IN PROGRESS. COMMAND IGNORED 'ASR'; RESULT: (SC2=02, SC1=00, MC=CMD0198)`
erscheint, wenn während der Systembeendigung noch Kommandos (SHOW-PENDING-MSG ausgenommen) eingegeben werden.

NBR0749 `COMMAND FROM /RUN FILE CANCELLED: 'RUN'; RESULT: (SC2=02, SC1=00, MC=NBR0749)`
erscheint, wenn ein Kommando aus der RUN-Datei zwar gelesen, aber wegen eines vorausgegangenen CANCEL-RUN-PROCESS-Kommandos nicht mehr ausgeführt wird.

### 16.5.2 Rückmeldungen an Bedienstationen

NBC0050 `CONSOLE ERROR - TRY AGAIN`
erscheint, wenn während des Lesens ein Hardwarefehler auftrat.

NBC0051 `TIMEOUT - INPUT IGNORED`
erscheint, wenn die vorhergehende physikalische Eingabeoperation wegen eines HW-Fehlers von der Software abgebrochen wird.

NBC0052 `INPUT NULL - IGNORED`
erscheint, wenn der Operator nur ein `END OF TEXT` eingegeben hat.

NBC0053 `RECIPIENT NOT AVAILABLE - INPUT IGNORED`
erscheint, wenn eine Eingabe nicht an den Empfänger (UCON/EMG-Handler) weitergereicht werden kann, da dieser nicht verfügbar ist.

NBC0054 `UCON NOT ACTIVE - INPUT IGNORED`
erscheint, wenn in der Zeit zwischen IPL-Ende und UCON-Start Kommandoeingaben oder Eingaben, die nicht mit `P.` beginnen, gemacht wurden.

NBC0055 `INVALID PREFIX ENTERED - NO SUCH ANSWER EXPECTED`
erscheint, wenn ein Eingabetext mit `P.` oder `E.` beginnt, aber keine Frage mit einem dieser Präfixe offen ist.

NBC0056 `HARDCOPY INOPERABLE`
erscheint, wenn der an einer Bedienstation vom Typ 3809/3886 angeschlossene Hardcopy-Drucker inoperabel ist. Je nach Kopplungsart ist auch das Display nicht mehr verfügbar.

NBC0057 `INPUT IGNORED - HARDCOPY DEFECTIVE`
erscheint, wenn bei eingestellter fester Kopplung zwischen Display und angeschlossenem Hardcopy-Drucker (nur bei Bedienstation 3809/3886) wegen eines Defektes beim Drucker keine Eingaben mehr akzeptiert werden.

NBC0058 `UCON BUSY - INPUT IGNORED`
erscheint, wenn eine Eingabe wegen UCON-Problemen nicht weitergegeben werden konnte.

NBC0059 `NO PF1-, PF2- OR ENTER-KEY USED - INPUT IGNORED`
erscheint bei Bedienstationen vom Typ 3809/3886, wenn eine Eingabe nicht mit der PF1-, PF2- oder ENTER-Taste abgeschlossen worden ist.

NBC0070 `CIO TESTS THIS CONSOLE`
erscheint, wenn das System sich aus einem Zustand ohne funktionsfähige Konsole erholt hat. Sind alle Bedienstationen defekt, so werden sie zyklisch mit dieser Meldung getestet. Die erste Bedienstation, die wieder operabel ist, gibt diesen Text aus.

### 16.5.3 Rückmeldungen in berechtigten Benutzerprogrammen und an Bedienstationen

␣␣REJ1 `TEXT NOT RECOGNIZED`

erscheint unter folgenden Bedingungen:

- wenn kein Empfänger für die Meldung angegeben ist
- wenn der Empfänger mit mehr als 4 Zeichen angegeben ist
- wenn nach dem Empfänger keines der Zeichen „/ % ? . : -“ und kein Leerzeichen folgt oder falls nach dem Empfänger mit „-“ ein Auftragskennzeichen eingeleitet wird, aber in der Zeichenfolge dahinter keines der folgenden Zeichen „/ % ? . :“ enthalten ist
- wenn nur ein „/“ oder nur ein „/“ und ein Kommandoauftragskennzeichen (cid) eingegeben werden

␣␣REJ5 `DESTINATION NOT FOUND`

erscheint, wenn eine Meldung einen falschen mnemotechnischen Gerätenamen für eine Bedienstation oder einen falschen Berechtigungsnamen enthält.

␣␣REJ6 `NO MATCH FOR REPLY`

erscheint, wenn die passende Frage zu der gesendeten Antwortmeldung nicht gefunden werden kann. Die Zurückweisung erfolgt auch dann, wenn versucht wird, eine Kommandozusatzinformationanforderung mit Punkt und eine Frage mit Doppelpunkt zu beantworten.
Meldungen des Meldungstyps „;“ können weder von Bedienstationen noch von berechtigten Benutzerprogrammen beantwortet werden, sie werden vom System beim Eintreffen eines erwarteten Ereignisses zurückgezogen.

␣␣REJ7 `USER NOT ALLOWED TO REPLY`

erscheint, wenn ein nicht berechtigter Benutzer versucht, auf eine Fragemeldung eine Antwortmeldung zu geben.

␣␣REJ8 `CONSOLE OFF - INPUT IGNORED`

erscheint, wenn sich die Bedienstation software-mäßig im Zustand SWITCHED OFF befindet (siehe Kommando CONSOLE, Handbuch „Kommandos“ [27]).

␣␣REJ9 `CONSOLE mn INOPERABLE - INPUT IGNORED`

erscheint, wenn die Bedienstation als defekt gekennzeichnet ist,
d.h. wenn das Hardcopy-Gerät ausgefallen ist und der Operand FORCED-HARDCOPY=YES gesetzt ist (siehe Kommando MODIFY-CONSOLE-OPTIONS, Handbuch „Kommandos“ [27]).

␣␣REJ10 `QUESTION IDENTIFICATION ALREADY EXISTING`

erscheint, wenn von einer Bedienstation eine Frage eingegeben wird, deren Nummer bereits einer anderen, von derselben Bedienstation gestellten und noch nicht beantworteten Frage, gegeben wurde.

␣␣REJ11 `MESSAGE CANNOT BE MATCHED TO AN ORDER`
erscheint, wenn sich ein berechtigtes Benutzerprogramm in seiner Eigenschaft als Kommandobearbeiter auf eine falsche Kommandoumgebung bezieht.

␣␣REJ12 `NBMHE/NBMAP ERROR IN PROTOCOL`
erscheint, wenn eine $CONSOLE-Anwendung, die mit V02 angeschlossen ist, eine Nachricht mit einem formal falschen NBMHE oder NBMAP sendet.

**i** Bei den Meldungen `REJ1` und `REJ5` bis `REJ7` wird zusätzlich gemeldet, welcher Text nicht erkannt, bzw. welche Meldung, Frage oder Antwort abgewiesen wurde.

Folgende Meldungen wurden durch Kommandoabschlussmeldungen ersetzt:

␣␣REJ2 `COMMAND NOT ALLOWED`
wird ersetzt durch die Kommandobeendigungsmeldung `NBR0742`.

NBR0742 `COMMAND NOT ALLOWED '...'; (RESULT: SC2=00, SC1=64, MC=CMD0216)`
erscheint, wenn die Bedienstation, von der aus das Kommando gegeben wurde, nicht berechtigt ist, dieses Kommando zu geben.

␣␣REJ3 `COMMAND NOT AVAILABLE`
wird ersetzt durch die Kommandobeendigungsmeldung `NBR0743`.

NBR0743 `COMMAND NOT AVAILABLE '...'; (RESULT: SC2=00, SC1=130, MC=CMD0200)`
erscheint, wenn die Routine, die das Kommando verarbeitet, nicht im System ist oder wenn zwischen der DCAM-Anwendung $CONSOLE und dem berechtigten Benutzerprogramm, das das Kommando ausführen soll, keine Verbindung besteht.

␣␣REJ4 `COMMAND NOT FOUND`
wird ersetzt durch die Kommandobeendigungsmeldung `NBR0744`.

NBR0744 `COMMAND NOT FOUND '...'; (RESULT: SC2=00, SC1=64, MC=NBR0744)`
erscheint, wenn ein Kommando eingegeben wurde, das kein Operator-Kommando ist.

## 16.6 Ablösung der Kommandos STATUS MSG und ASR

Die Kommandos STATUS MSG und ASR wurden bereits in BS2000/OSD-BC V3.0 durch die folgenden Kommandos abgelöst, werden jedoch aus Kompatibilitätsgründen weiterhin unterstützt.
Das Kommando ASR steht dem Operator nur noch im eingeschränkten Funktionsumfang und an der Hauptbedienstation nicht mehr automatisch zur Verfügung. Mit dem ASR-Kommando können Berechtigungsschlüssel (Routing-Codes) auch nur noch an Anwendungen mit generiertem Berechtigungsnamen vergeben werden.

| **Kommando STATUS** | **Ersatz-Kommando** | **BS** |
|---|---|---|
| MSG | SHOW-PENDING-MSG | @ |
| MSG,<br>ALL | SHOW-PENDING-MSG DESTINATION=*ANY | @ |

| **Kommando ASR** | **Ersatz-Kommando** | **BS** |
|---|---|---|
| ADD,<br>CODE=... | REQUEST-OPERATOR-ROLE | @ |
| ADD,<br>CODE=...,<br>FILTER=.... | ADD-CONSOLE-FILTER FILTER=...,ROUTING-CODE=... | @ |
| ADD,<br>CODE=...,<br>FILTER=ALL | ADD-CONSOLE-FILTER FILTER=(1,2,3,4),-<br>ROUTING-CODE=... | @ |
| ADD,<br>CONSOLE=... | hat kein Ersatz-Kommando | -- |
| DELETE,<br>CODE=... | RELEASE-OPERATOR-ROLE | @ |
| DELETE,<br>CODE=...,<br>FILTER=... | REMOVE-CONSOLE-FILTER FILTER=...,-<br>ROUTING-CODE=... | @ |
| DELETE,<br>CODE=...,<br>FILTER=ALL | REMOVE-CONSOLE-FILTER FILTER=(1,2,3,4),-<br>ROUTING-CODE=... | @ |
| DELETE,<br>CONSOLE=... | hat kein Ersatz-Kommando | -- |
| DESTINATION | SHOW-SYSTEM-PARAMETERS PARAM=MSGDEST | R |

| **Kommando ASR** | **Ersatz-Kommando** | **BS** |
|---|---|---|
| HELP | SHOW-CONSOLE-STATUS CONSOLE=*OWN | @ |
| HELP,<br>CODE=... | SHOW-CONSOLE-STATUS CONSOLE=*ALL [1] | @ |
| HELP,<br>CODE=...,<br>FILTER=... | SHOW-CONSOLE-FILTER SELECT=*FILTER(...)<br>oder<br>SHOW-CONSOLE-FILTER SELECT=*ROUTING-CODE(...) | @ |
| HELP,<br>COMMAND=.... | SHOW-CMD-ATTRIBUTES | @ |
| HELP,<br>CONSOLE=... | SHOW-CONSOLE-STATUS CONSOLE=... | @ |
| HELP,<br>CONSOLE=ALL | SHOW-CONSOLE-STATUS CONSOLE=*ALL | @ |
| HELP,<br>CONSOLE=...,<br>FILTER=... | SHOW-CONSOLE-FILTER SELECT=*FILTER(...) [2] | @ |
| INF | REMOVE-CONSOLE-FILTER FILTER=*ALL,-<br>ROUTING-CODE=*ALL<br>oder<br>MODIFY-MSG-SUBSCRIPTION DELIVER-OTHER-MSG=-<br>*YES | @<br><br><br>E |
| MAIN | REQUEST-MAIN-CONSOLE-FUNCTIONS | E |
| NOINF | ADD-CONSOLE-FILTER FILTER=ALL,-<br>ROUTING-CODE=*ALL<br>oder<br>MODIFY-MSG-SUBSCRIPTION DELIVER-OTHER-MSG=-<br>*NO | @<br><br><br>E |
| PRIMARY | hat kein Ersatz-Kommando | -- |
| SUPPRESS | RESET-MSG-BUFFER SENDER=... | @ |

[1] wird nicht direkt abgebildet, die Information ist aber im angegebenen Kommando enthalten

[2] nur bezogen auf die eingebende Bedienstation

# 17 Automatisierung von Operatorfunktionen

## 17.1 Berechtigte Benutzerprogramme mit Operatorfunktionen

Mit der $CONSOLE-Schnittstelle von BS2000/OSD kann sich der Operator von solchen Arbeiten befreien, die sich programmgesteuert ausführen lassen, z.B.

- regelmäßiges Abfragen bestimmter Lastwerte des Systems (z.B. Anzahl von Jobs nach unterschiedlichen Kriterien, Auslastung des Hauptspeichers);
  Veranlassen geeigneter Maßnahmen, falls erforderlich
- regelmäßiges Prüfen, ob unbeantwortete Meldungen ausstehen; Nachrichten an die Bedienstationen senden
- Geräteverwaltung

Tätigkeiten dieser Art lassen sich auf Benutzerprogramme übertragen, die sich als BCAM-/DCAM-Anwendungen dem Operating-System gegenüber als berechtigte $CONSOLE-Anwendungen ausgewiesen haben und als logische Bedienstationen fungieren. Diese als Bedienstation fungierenden Benutzerprogramme werden im Allgemeinen als **berechtigte Benutzerprogramme** bezeichnet. Sie sind berechtigt, Operatorfunktionen entsprechend ihrem Berechtigungsprofil (Routing-Code-Menge) auszuführen.

Ein unberechtigtes Benutzerprogramm darf lediglich Meldungen an das System senden und ggf. Antworten erhalten. Es handelt sich dabei um eine BCAM-/DCAM-Anwendung, die sich zwar an $CONSOLE anschließt, aber keinen speziellen (berechtigten) Verbindungsaufbau beantragt hat.

Berechtigte Programme können alle herkömmlichen Aktivitäten eines Operators ausführen.
Darüber hinaus kann ein berechtigtes Programm als Kommandoausführungsinstanz für Operator-Spezialkommandos auftreten (mit der Parameter-Anweisung ADD-CMD-ENTRY oder durch das Kommando CONNECT-CMD-SERVER definiert), siehe Seite 684.

Bei der Anmeldung eines eigenen Operator-Kommandos mit CONNECT-CMD-SERVER ist die Angabe COMPLETION-CONTROL=*YES möglich. Dabei ist zu beachten, dass – im Unterschied zu Konsolen – die Kommandoeingabe in einer Benutzertask synchron erfolgt.

Die Eingabe ist also erst nach beendeter Ausführung des vorherigen Kommandos möglich. Deshalb sollte das Kommando entweder eine kurze Bearbeitungsdauer aufweisen oder mit COMPLETION-CONTROL=*NO angemeldet werden.



Vom Hersteller werden auf dieser Schnittstelle basierende Produkte mit eigener Oberfläche angeboten, z.B. OMNIS und PROP-XT (siehe Seite 693), MAREN (siehe „MAREN“ [31]) und ROBAR (siehe „ROBAR“ [43]).

Im Betriebsmodus NBCONOPI=Y besitzt die Operator-Task für ein berechtigtes Benutzerprogramm mit dynamischem Berechtigungsnamen die Privilegien seines Benutzers und das Privileg OPERATING.

Für berechtigte Benutzerprogramme gilt dieselbe Einteilung der Systembedienung in Aufgabengebiete wie für Bedienstationen.
Jedes Benutzerprogramm, das Operatorfunktionen ausüben will, hat sich mit der zentralen Task der Systembedienung (unter dem Applikationsnamen $CONSOLE) zu verbinden, zu „konnektieren“. Falls Operatorfunktionen ausgeübt werden sollen, die eine Berechtigung (Berechtigungsschlüssel, Routing-Code) voraussetzen, muss das Benutzerprogramm auch seine Berechtigung für diese Konnektierung nachweisen.

Die Verknüpfung eines Benutzerprogramms mit einem **Berechtigungsnamen** verleiht diesem Programm ein Merkmal, das mit dem mnemotechnischen Gerätenamen einer Bedienstation vergleichbar ist.

Für berechtigte Benutzerprogramme, die sich anhand dieses Berechtigungsnamens an das System anschließen (generierter Berechtigungsname), muss im OPR-Parameterservice mit SET-CODE bzw. ADD-CMD-ENTRY ein Eintrag in die Berechtigungsnamentabelle erzeugt werden. Ein Berechtigungsname besteht aus 4 alphanumerischen Zeichen (Werte: A-Z, 0-9, oder @, $, #), wobei das erste Zeichen keine Ziffer und kein „#“ sein darf. „@“ als erstes Zeichen sollte nicht verwendet werden.
Vom System werden immer 192 Berechtigungsnamen erzeugt. Im Parameterservice können maximal 64 Berechtigungsnamen (generierte Berechtigungsnamen) angegeben werden. Die restlichen Berechtigungsnamen werden vom System vergeben (dynamische Berechtigungsnamen, @001 bis @192, sofern noch nicht vergeben).

Auf diese Weise lassen sich berechtigte Benutzerprogramme mit Namen bezeichnen, die die gleiche Lebensdauer haben wie die mnemotechnischen Gerätenamen der Bedienstationen.

Berechtigungsnamen haben folgende Eigenschaften:

- Die Lebensdauer von Berechtigungsnamen ist gleich der Lebensdauer des Systems und damit gleich der Lebensdauer der Gerätedefinitionen für die Bedienstationen.
- Sobald ein Benutzerprogramm mit einem Berechtigungsnamen verknüpft ist, gilt es als berechtigtes Benutzerprogramm. Die Verknüpfung ist vom Programm zu veranlassen und gilt, bis das Benutzerprogramm die Verknüpfung aufhebt, bis das Programm beendet wird oder bis UCON die Verbindung abbricht.

- Die Zuordnung von Aufgabengebieten für berechtigte Benutzerprogramme lässt sich, abhängig von der Anschlussart, entweder über den Berechtigungsnamen oder über Operator-Rollen herstellen.
- Nachdem ein Anschluss vom System akzeptiert wurde, ist ein berechtigtes Benutzerprogramm nur am Berechtigungsnamen als Sender oder als Empfänger von Nachrichten zu erkennen.

Einem Berechtigungsnamen kann eine Kommandobearbeitungsfunktion zugeordnet werden. Soll ein berechtigtes Benutzerprogramm Kommandos ausführen, so muss dem System bekannt sein, für welche Kommandos es als Kommandobearbeiter (Kommando-Server für Spezial-Kommandos) fungiert. Für berechtigte Anwendungen mit generiertem Berechtigungsnamen erfolgt die Zuordnung Kommandoname/Berechtigungsname im OPR-Parameterservice mit dem Satz ADD-CMD-ENTRY.
Daneben steht **allen** berechtigten Benutzerprogrammen zur Verknüpfung von berechtigten Anwendungen mit Operatorkommandos das Kommando CONNECT-CMD-SERVER zur Verfügung. Zum Lösen der Verknüpfung dient das Kommando DISCONNECT-CMD-SERVER (siehe Handbuch „Kommandos“ [27]).
Beide Kommandos sind standardmäßig mit dem Berechtigungsschlüssel K geschützt und nur für berechtigte Benutzerprogramme zulässig.

Die Systembetreuung erzeugt für jedes berechtigte Benutzerprogramm, das sich an das System anschließen darf, einen Eintrag im Benutzerkatalog. Sie sollte dabei Kennwörter zum Schutz vor unbefugter Aufnahme einer Verbindung zur Anwendung $CONSOLE (UCON) vereinbaren. Die Einträge in den Benutzerkatalog erfolgen mit dem Kommando ADD-USER; als Benutzerkennung wird bei Anwendungen mit generiertem Berechtigungsnamen der Berechtigungsname, bei Anwendungen mit dynamischem Berechtigungsnamen die OPERID verwendet.

*Beispiel für einen Eintrag in den Benutzerkatalog*

```
/ADD-USER USER-ID=RUDI,PROTECTION-ATTRIBUTES=(-
LOGON-PASSWORD=C'FOX#HOLE'),ACCOUNT-ATTRIBUTES=(ACCOUNT=K0815)
```

Als Benutzerkennung wird für Benutzerprogramme mit generiertem Berechtigungsnamen der Berechtigungsname verwendet.

```
/ADD-USER USER-ID=ISOLDE,PROTECTION-ATTRIBUTES=(-
LOGON-PASSWORD=C'MUSTARD!'),ACCOUNT-ATTRIBUTES=(ACCOUNT=K0815)
```

Als Benutzerkennung wird für Benutzerprogramme mit dynamischem Berechtigungsnamen die OPERID verwendet.

Ein sedezimales Kennwort ist nicht erlaubt.



Ein berechtigtes Benutzerprogramm darf prinzipiell jedes Operatorkommando verwenden; davon ausgenommen sind nur die Kommandos, deren Eingabeort eigenen Einschränkungen unterliegt (z.B. REQUEST-MAIN-CONSOLE-FUNCTIONS). Ein berechtigtes Benutzerprogramm kann nicht Ersatz- oder Hauptbedienstation sein.

Ist in der zugeteilten Operator-Rolle die Berechtigung für das Kommando CONNECT-CMD-SERVER enthalten – standardmäßig ist das der Berechtigungsschlüssel K – so kann sich das berechtigte Benutzerprogramm mit diesem Kommando für die Bearbeitung von Operatorkommandos zuständig erklären.
Das System reicht Operatorkommandos, für deren Ausführung sich ein berechtigtes Benutzerprogramm zuständig erklärt hat, ohne syntaktische Analyse an das Programm weiter.



**ACHTUNG!**
Inhaber des Berechtigungsschlüssels K könnten versehentlich wichtige Operatorkommandos vereinnahmen und z.B. außer Kraft setzen. Der Systembetreuer sollte deshalb diese Berechtigung nur an ausgewählte Benutzerprozesse vergeben.

### Aufbau der Berechtigungsnamentabelle

Den maximal 64, im Startup-Parameterservice mit dem Parametersatz OPR erzeugten, Berechtigungsnamen (generierte Berechtigungsnamen) werden vom System für berechtigte Benutzerprogramme, die sich anhand einer Operator-Identifikation anschließen, solange eindeutige Berechtigungsnamen hinzugefügt, bis die maximale Anzahl von 192 Einträgen erreicht ist.

Die Systembetreuung definiert mit dem Kommando CREATE-OPERATOR-ROLE die Liste von Berechtigungsschlüsseln, die zu einer Operator-Rolle zusammengefasst und dem angegebenen Pubset bekannt gemacht werden. Mit dem Kommando MODIFY-OPERATOR-ATTRIBUTES wird die Zuordnung von Operator-Rollen zu einer Operator-Identifikation hergestellt bzw. geändert.

Die Konnektierung als berechtigtes Benutzerprogramm kann auf zwei verschiedene Arten erfolgen:

- als Anschluss mit generiertem Berechtigungsnamen (Benutzerprogramme alter Art); siehe Seite 668
- als Anschluss mit dynamischem Berechtigungsnamen; siehe Seite 668

| **Kommando** | **Bedeutung** |
|---|---|
| ADD-USER | Einträge in den Benutzerkatalog |
| CREATE-OPERATOR-ROLE | Eine oder mehrere Operator-Rollen generieren |
| DELETE-OPERATOR-ROLE | Löscht die Definition einer Operator-Rolle |
| EXIT-JOB | Beenden der $CONSOLE-Anwendung inkl. Verbindungsabbau |
| MODIFY-OPERATOR-ATTRIBUTES | Zuordnung von Operator-Rollen zu einer Operator-Identifikation herstellen bzw. ändern |
| RELEASE-OPERATOR-ROLE | Beantragt die Rückgabe eines spezifischen Tätigkeitsbereichs |
| REQUEST-OPERATOR-ROLE | Zuweisung einer oder mehrerer Operator-Rollen beantragen |
| SHOW-OPERATOR-ROLE | Zeigt Informationen über Operator-Rollen an |
| **Makro** | **Bedeutung** |
| NBMAP | Nachspann der Nachricht beschreiben (siehe Seite 683) |
| NBMHE | Format des Nachrichtenheaders beschreiben (siehe Seite 680) |

Tabelle 50: Schnittstellenübersicht zu „Berechtigten Benutzerprogrammen mit Operatorfunktionen“

### 17.1.1 Anschlüsse mit generierten Berechtigungsnamen

Die Aufgabengebiete – und damit die Berechtigungen – können einem berechtigten Benutzerprogramm mit generiertem Berechtigungsnamen auf zwei Arten zugeordnet werden:

- durch Voreinstellung (Startup-Parameterservice, Parametersatz OPR))
- dynamisch durch das Kommando ASR

Nach Abbruch einer Verbindung bleiben die Berechtigungen dem generierten Berechtigungsnamen erhalten, so dass eine neue Konnektierung unter diesem Namen alle diese Berechtigungen erbt.

$CONSOLE-Anwendungen mit generierten Berechtigungsnamen werden nur noch aus Kompatibilitätsgründen unterstützt. Der Verbindungsaufbau funktioniert noch unverändert; da aber eine Neuimplementierung von $CONSOLE-Anwendungen mit generierten Berechtigungsnamen nicht erwünscht ist, wird er nicht mehr beschrieben.

### 17.1.2 Anschlüsse mit dynamischen Berechtigungsnamen

Anschlüsse mit generiertem Berechtigungsnamen genügen nicht mehr den strengen Sicherheitsanforderungen an ein System, da die Aufgabengebiete, d.h. eine Summe von Berechtigungsschlüsseln, immer nur einem Gerät oder einem berechtigten Benutzerprogramm, nicht aber einer Person zugeordnet werden können.

Diese Personalisierung wird bei Anschlüssen mit dynamischen Berechtigungsnamen gewährleistet, da bei der Konnektierung eine Operator-Identifikation (entspricht einer Benutzerkennung) angegeben werden muss. Unmittelbar nach der Konnektierung verfügt das berechtigte Benutzerprogramm über keinerlei Berechtigungen, kann also nur Operatorfunktionen, die keine Berechtigungen voraussetzen, und Kommandos mit dem Berechtigungsschlüssel @ nutzen. Erst mit dem Kommando REQUEST-OPERATOR-ROLE (Berechtigungsschlüssel @) können Berechtigungen beantragt werden, wobei die Vergabe einer OPERID-bezogenen Prüfung unterliegt.
Eine Operator-Rolle entspricht einem Aufgabengebiet und ist eine von der Systembetreuung festgelegte Menge von Berechtigungsschlüsseln, wobei beliebige Kombinationen der insgesamt 40 Berechtigungsschlüssel möglich sind. Ein berechtigtes Benutzerprogramm kann mehrere Operator-Rollen beantragen. Die logische Summe der Operator-Rollen ergibt dann das Berechtigungsprofil eines berechtigten Benutzerprogramms.

Mit der Festlegung von Operator-Rollen durch die Systembetreuung können auch Operatorfunktionen **personenbezogen** aufgeteilt und berechtigten Benutzerprogrammen zugewiesen werden.

#### Verbindungsaufbau mit dynamischem Berechtigungsnamen

Der Verbindungsaufbau einer berechtigten Anwendung mit dynamischem Berechtigungsnamen erfolgt grundsätzlich in mehreren Schritten. Die meisten davon sind nur optional, abhängig von den in der Verbindungsnachricht gemachten Angaben und davon, wie die verwendete Operator-Kennung geschützt ist.

Von $CONSOLE (Systemtask UCON) kann nicht überprüft werden, ob die Angabe $CONSOLE oder @CONSOLE in der ankommenden Verbindungsanforderung der Realität entspricht ($CONSOLE für die Datensichtstation als Endbenutzer, @CONSOLE für ein Programm als Endbenutzer), da diese Verbindungsanforderung immer von einem Vermittlerprogramm (z.B. OMNIS) eingereicht wird.
Der Systembetreuer sollte darauf achten, dass nur solche Programme als Vermittler eingesetzt werden, die die Korrektheit der Eingabe (erstes Zeichen „$“ oder „@“) sicherstellen. Er kann ausgewählten Kennungen den Zugang zu BS2000/OSD ausschließlich von der Datensichtstation erlauben (siehe Kommando SET-LOGON-PROTECTION).

1. Schritt: Verbindungaufforderung

   Das berechtigte Benutzerprogramm sendet zunächst an $CONSOLE des gewünschten Systems eine Verbindungsanforderung mit folgender Struktur (x=$ oder x=@, je nach Art des Programmanschlusses, siehe auch Abschnitt „Format der Verbindungsnachricht für dynamische Berechtigungsnamen“ auf Seite 672):
   `NAME=xCONSOLE[,OPERID=<name>][,PASSWORD=<password>][,PROTVERS=V<integer>]`

   Der Anschluss wird **vorläufig** akzeptiert, wenn folgende Kriterien erfüllt sind:

   – es sind nicht alle zu vergebenden Berechtigungsnamen belegt
   – der Systemparameter NBBAPRIV erlaubt die Anmeldung

   Mit der vorläufigen Akzeptierung wird ein Begleittext gesendet, dem die von $CONSOLE (UCON) akzeptierte Protokollversion (PROTVERS) und die BS2000-Version zu entnehmen sind. Wenn die beantragte Protokollversion größer als die höchste Version ist, die von $CONSOLE unterstützt wird, enthält der Begleittext den maximalen Wert, den UCON akzeptieren kann.

   Struktur des Begleittextes:

| | | |
|---|---|---|
| 00 - 18 DC C'CONNECTION ACCEPTED' | | 19 Byte |
| 19 DC C' ' | Leerzeichen | 1 Byte |
| 20 - 21 DC C'nn' | Protokollversionsnummer | 2 Byte |
| 22 DC C',' | Komma | 1 Byte |
| 23 - 26 DC C'nn.n' | BS2000-Versionsnummer | 4 Byts |

   Für BS2000/OSD-BC V9.0 wird als BS2000-Versionsnummer C'18.0' ausgegeben.

Unabhängig von der in der Verbindungsanforderung beantragten Version des Protokolls, werden ab diesem Zeitpunkt, bis zur endgültigen Akzeptierung oder Ablehnung der Verbindung, alle Nachrichten in beide Richtungen mit einem Header versehen. Das Protokoll muss strikt eingehalten werden; unerwartete Eingaben während des Anmeldungsdialogs führen zur Zurückweisung des Konnektierungsversuches.

2. Schritt: OPR-LOGON-REQUEST

   Wenn der Operand OPERID in der Verbindungsnachricht fehlt, wird er vom System durch die Nachricht `PLEASE LOGON` angefordert.
   Das berechtigte Benutzerprogramm hat nun mit der Nachricht zu antworten (die ISP-Form `LOGON <name>,,<kennwort>` ist zulässig):
   `SET-LOGON-PARAMETERS USER-ID=<name>[,PASSWORD=<password>]`

   Das im Kommando SET-LOGON-PARAMETERS angegebene Kennwort wird für die weiteren Überprüfungen benutzt. Das in der Verbindungsnachricht eventuell angegebene Kennwort wird grundsätzlich ignoriert, wenn ein nachträgliches Kommando SET-LOGON-PARAMETERS angefordert wurde.

3. Schritt: Prüfung der Schutzkriterien

   Anhand der bis hierher erhaltenen Information werden vom System die im Benutzerkatalog hinterlegten Schutzkriterien der Operator-Identifikation abgefragt.

   Wird bei der Überprüfung der Authentisierungsdaten festgestellt, dass für diese Operator-Identifikation ein Anschluss in der angegebenen Betriebsart (Attribute OPERATOR-ACCESS-TERMINAL bzw. OPERATOR-ACCESS-PROGRAM im Benutzerkatalog) nicht erlaubt ist, wird die Verbindung wieder abgebaut.

   Wird bei der Überprüfung der Authentisierungsdaten festgestellt, dass der Anschluss nicht durch Kennwort geschützt ist, wird der Schritt 4 übersprungen.

4. Schritt: Anforderung des Kennworts

   Wenn das Kennwort bis jetzt noch nicht angegeben worden ist, wird es vom System mit der Nachricht `PLEASE ENTER PASSWORD` angefordert.
   Das berechtigte Benutzerprogramm hat nun mit dem Kennwort zu antworten.
   Unter OMNIS erfolgt die Eingabe dunkelgesteuert.

5. Schritt: Prüfung der Zugangsberechtigung

   Nachdem das berechtigte Benutzerprogramm alle notwendigen Daten an das System weitergeleitet hat, muss es noch die Bestätigung des Anschlusses an das System abwarten, bevor es mit dem normalen Nachrichtenverkehr, gemäß der in der Verbindungsanforderung geforderten Protokollversion, beginnen kann.

   Das System überprüft die Anschlussdaten und akzeptiert **endgültig** den Anschluss, wenn das angegebene Kennwort korrekt ist bzw. kein Kennwort erforderlich ist und OPERID und PASSWORD im Benutzerkatalogeintrag zusammenpassen (wenn die OPERID mit Kennwort zu authentisieren ist).

Die Bestätigung enthält folgenden, an der Datensichtstation auszugebenden Text:

```
CONNECTION REQUEST ACCEPTED, APPLICATION NAME = @xxx
```

Das angeschlossene berechtigte Benutzerprogramm kann zunächst nur einige nicht schützenswerte Kommandos eingeben (Schloss @). Um Operatorfunktionen auszuführen, muss es mit dem Kommando REQUEST-OPERATOR-ROLE eine oder mehrere Operator-Rollen beantragen.

### Format der Verbindungsnachricht für dynamische Berechtigungsnamen

```
[NAME=]xCONSOLE [,[OPERID=]name]

[,[PASSWORD=]C'password 1..8']

[,[PROTVERS=]Vn]

[,[DISCON=]{YES/NO}]
```

Die Verbindungsanforderung kann auch ohne Schlüsselworte angegeben werden.

**xCONSOLE**
kennzeichnet die Art des Programmanschlusses im sicheren System
(siehe Kommando SET-LOGON-PROTECTION im SECOS-Handbuch „Zugangs- und Zugriffskontrolle“ [48]). Für **x** kann „$“ oder „@“ eingesetzt werden:

**$CONSOLE**
zeigt an, dass das berechtigte Benutzerprogramm im Dialog mit einer Person arbeitet.
Die Prüfung der Zugangsberechtigung erfolgt gegen die mit SET-/MODIFY-LOGON-PROTECTION, Operand OPERATOR-ACCESS-TERM, getroffenen Festlegungen.

**@CONSOLE**
zeigt an, dass das berechtigte Benutzerprogramm im Batch-Betrieb arbeitet.
Die Prüfung der Zugangsberechtigung erfolgt gegen die mit SET-/MODIFY-LOGON-PROTECTION, Operand OPERATOR-ACCESS-PROGRAM, getroffenen Festlegungen.

**OPERID=<name>**
Name einer Operator-Identfikation. Eine Operator-Identifikation ist eine Benutzerkennung für Operatoren. Einen funktionellen Unterschied zu einer gewöhnlichen Benutzerkennung gibt es zunächst nicht. Erst wenn mit dem Kommando SET-LOGON-PROTECTION einer Benutzerkennung Operatorberechtigungen zugeordnet wurden, kann man sinnvollerweise von einer Operator-Identifikation sprechen.

**PASSWORD=C'<password 1..8>'**
Für eine Operator-Identfikation kann (wie für jede Benutzerkennung) ein Kennwort vereinbart werden. Das im Benutzerkatalog einer Operator-Identfikation eingetragene Kennwort ist anzugeben.

Ein sedezimales Kennwort ist nicht erlaubt. Das Kennwort darf max. 8 Zeichen lang sein.

**PROTVERS=Vn**
bezeichnet die Versionsnummer der Schnittstelle zwischen der Anwendung und $CONSOLE (n = 00, 01, 02). Jede Versionsnummer entspricht einem eindeutigen Kommunikationsprotokoll, das von $CONSOLE unterstützt und eingehalten wird.

Der Standardwert ist 00. In diesem Fall erfolgt der Nachrichtenverkehr zwischen System und berechtigtem Benutzerprogramm in beiden Richtungen ohne einen Header.

Bei der Version 01 empfängt das berechtigte Benutzerprogramm vom System die Nachricht und einen Header. Der Nachricht folgt bei Meldungen aus der Meldungsdatei, die mit dem Makro MSG7 oder MSG7X ausgegeben werden, ein Nachspann, der die Nachricht zur programmierbaren Auswertung „datenorientiert" enthält. Die Datenstruktur des Headers wird vom Makro NBMHE festgelegt, die des Nachspanns vom Makro NBMAP (siehe Seite 680 und Seite 683). Der Nachrichtenverkehr vom berechtigten Benutzerprogramm zum System wird, wie bei der Protokollversion 00, ohne Header durchgeführt.

Bei der Version 02 wird der Nachrichtenverkehr in beiden Richtungen mit Header durchgeführt. Berechtigte Benutzerprogramme sind damit in der Lage, Nachrichten als Kommandoergebnisse und Zusatzinformationen zu kennzeichnen und dem System das Ende einer Kommandobearbeitung mitzuteilen.

Fehlt in der Verbindungsnachricht der Operand PROTVERS, so wird der Standardwert 00 eingesetzt. Wurde in PROTVERS ein zu hoher Wert angegeben, so wird die höchste vom System bediente Version im Begleittext der Verbindungsakzeptierung eingetragen. Die beantragte Versionsnummer der Schnittstelle kann also herabgesetzt werden. Ist das berechtigte Benutzerprogramm mit der herabgesetzten Versionsnummer nicht einverstanden, kann es die Verbindung abbauen.

**DISCON**
legt fest, wie das System reagiert, wenn das berechtigte Benutzerprogamm mit dem Empfang von Meldungen so weit in den Rückstand gerät, dass die durch den Systemparameter NBRCSCK[N] festgelegte Zeit überschritten wird.

**DISCON=YES**
Voreinstellung: Das System baut die Verbindung zum berechtigten Benutzerprogramm ab.

**DISCON=NO**
Das berechtigte Benutzerprogramm wird nicht diskonnektiert. Statt dessen werden alle Meldungen, die zur Ausgabe an das berechtigte Benutzerprogramm anstehen, gelöscht und durch eine einzelne Meldung `NBR0601` ersetzt.

```
NBR0601 SOME MESSAGES DISCARDED SINCE THE DESTINATION APPLICATION DID NOT
        RECEIVE THEM JUST IN TIME
```

### Verbindungsabbau

Zum Beenden der $CONSOLE-Anwendung einschließlich Verbindungsabbau wird (mit NBCONOPI=Y) das Kommando EXIT-JOB angeboten. Damit wird implizit auch die Funktion von RELEASE-OPERATOR-ROLE OPERATOR-ROLE=*ALL realisiert und damit die Rückgabe aller Berechtigungsschlüssel.

### 17.1.3 Austausch von Nachrichten

Jedes berechtigte Benutzerprogramm tauscht seine Nachrichten über die DCAM-Anwendungen aus. DCAM bietet Schnittstellen für die Programmiersprachen COBOL und Assembler an:

- Wenn das zu lösende Problem keine besonderen Anforderungen an die Übermittlung von Nachrichten stellt, so lässt sich COBOL einsetzen.
- Ist aber eine asynchrone Verarbeitung eintreffender Meldungen erforderlich, so ist in Assembler (mit Makroaufrufen zur ereignisgesteuerten Verarbeitung) zu programmieren. Einzelheiten sind den Handbüchern „DCAM-Programmschnittstellen“ [13], „DCAM-Cobol-Aufrufe“ [11], „DCAM-Makroaufrufe“ [12] und „Makroaufrufe an den Ablaufteil“ [30] zu entnehmen.

#### Grobstruktur eines berechtigten Benutzerprogramms

Das folgende Bild gibt zunächst eine Übersicht, wie ein berechtigtes Benutzerprogramm Nachrichten mit dem System austauscht.

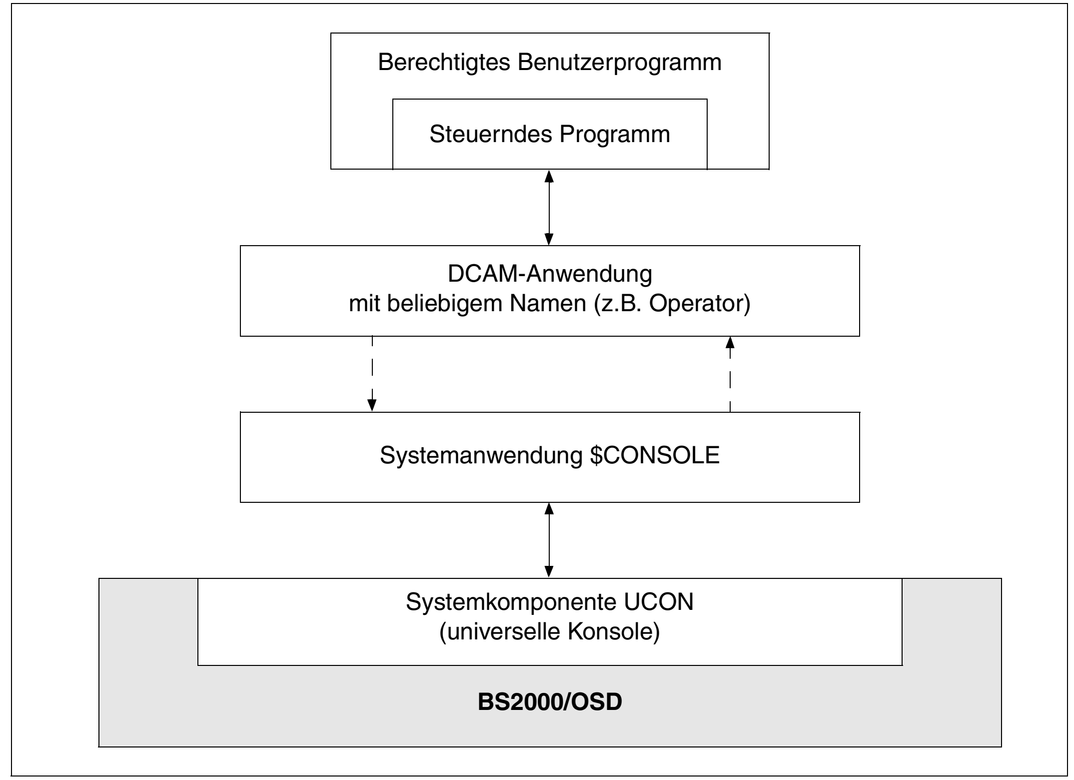



Bild 34: Nachrichtenaustausch zwischen BS2000/OSD und berechtigtem Benutzerprogramm

Da jedes berechtigte Benutzerprogramm seine Nachrichten über DCAM-Anwendungen mit dem System austauscht, ergeben sich zwei Grobstrukturen für solche Programme: synchrone und asynchrone Verarbeitung.

- Synchrone Verarbeitung

  Die Grobstruktur für ein Programm, das eintreffende Meldungen synchron verarbeitet, zeigt das folgende Bild. Dabei kann der Inhalt der Schleife je nach gegebener Aufgabenstellung anders angeordnet sein.

  In Klammern ist jeweils der zu verwendende COBOL- bzw. Makroaufruf angegeben.

| Anfang | | |
|---|---|---|
| Eröffnen einer DCAM-Anwendung (YOPEN) | | |
| Aufbauen einer logischen Verbindung zur Systemanwendung $CONSOLE (YOPNCON):<br>gleichzeitig Nachweisen der Berechtigung (siehe die verschiedenen Verbindungsaufbau-beschreibungen)<br>Ende des Verbindungsaufbaues | | |
| Vorbereiten der Verarbeitung | | |
| SCHLEIFE, die bis zum Eintreten einer Endebedingung durchlaufen wird | | |
| | SCHLEIFE | |
| | | Empfangen von Nachrichten bis eine Nachricht erkannt wird, die für dieses Benutzerprogramm bedeutsam ist (YRECEIVE) [1] |
| | Verarbeiten der zuletzt empfangenen Meldung: Senden einer Antwort, falls nötig (YSEND) | |
| Abbauen der Verbindung zur Systemanwendung $CONSOLE (YCLSCON) | | |
| Schließen der DCAM-Anwendung (YCLOSE) | | |
| Ende | | |

[1] Wird ein Berechtigungsschlüssel nicht benötigt (weder als Kommandoschlüssel noch als Routing-Code), ist es von Vorteil, ihn nicht zu besetzen. Das Transportieren von Meldungen, die vom Empfänger ignoriert werden, verringert die Performance und kann zu Systemengpässen führen.

Bild 35: Grobstruktur für ein berechtigtes Benutzerprogramm (Meldungen synchron verarbeiten)

- Asynchrone Verarbeitung

  Die Grobstruktur für ein Programm, das eintreffende Meldungen asynchron verarbeitet, unterscheidet sich von der vorstehenden im Wesentlichen durch den Inhalt der Schleife und eine geringe Änderung im Schritt „Vorbereiten der Verarbeitung“: Ereigniskennungen sind einzurichten und Contingency-Definitionen dem System bekanntzugeben; siehe Makroaufrufe ENAEI und ENACO im Handbuch „Makroaufrufe an den Ablaufteil“ [30]).
  Die Schleife enthält hier nur die Anforderung einer Nachricht; wenn die Nachricht eintrifft, lässt sie sich in einem Contingency-Prozess verarbeiten, das Senden einer Antwort eingeschlossen.

In die Schleife lassen sich weitere Tätigkeiten einbeziehen, die nicht zum Verarbeiten von Nachrichten gehören. Das asynchrone Verarbeiten von Nachrichten ist deshalb bei Aufgabenstellungen sinnvoll, die zusätzliche Koordinierungstätigkeiten erfordern oder bei denen mehrere Kommunikationspartner zu bedienen sind.

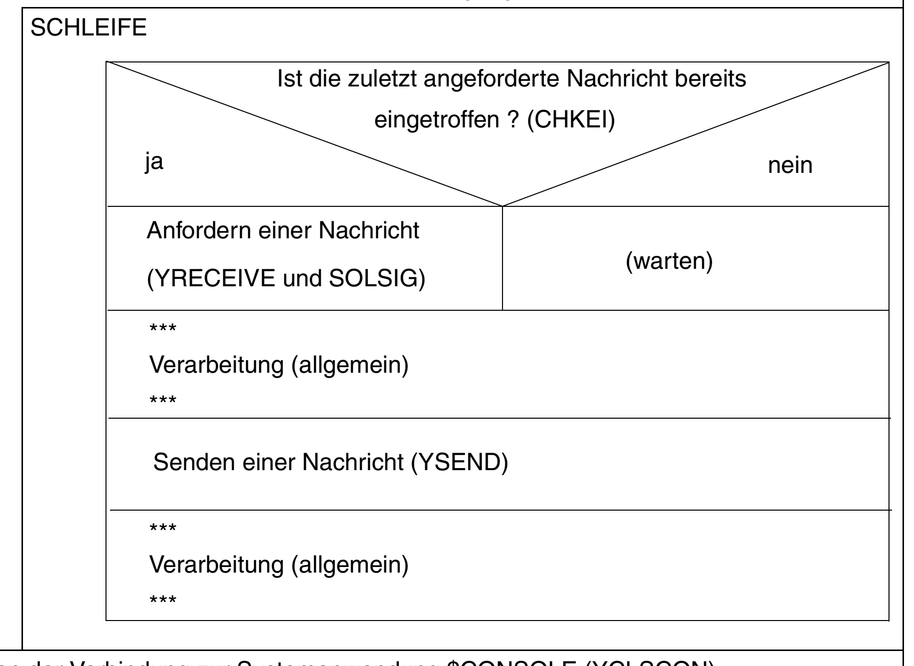
Anfang

Eröffnen einer DCAM-Anwendung (YOPEN)

Aufbauen einer logischen Verbindung zur Systemanwendung $CONSOLE (YOPNCON): gleichzeitig Nachweisen der Berechtigung (siehe die verschiedenen Verbindungsaufbaubeschreibungen)
Ende des Verbindungsaufbaues

Vorbereiten der Verarbeitung:
Ereigniskennungen sind einzurichten und Contingency-Definitionen dem System bekanntzugeben

SCHLEIFE, die bis zum Eintreten einer Endebedingung durchlaufen wird




Abbauen der Verbindung zur Systemanwendung $CONSOLE (YCLSCON)

Schließen der DCAM-Anwendung (YCLOSE)

Ende

Bild 36: Grobstruktur für ein berechtigtes Benutzerprogramm (Meldungen asynchron verarbeiten)

### Überwachung der Abnahme von Nachrichten

Von der UCON-Task wird zyklisch (jede Minute) überprüft, ob berechtigte Benutzerprogramme die an sie verschickten Nachrichten entgegennehmen. Wird festgestellt, dass die nächste zur Ausgabe anstehende Meldung schon länger als eine Minute auf den Empfang durch das berechtigte Benutzerprogramm wartet, wird das mit der Meldung `NBR0600` an der Bedienstation gemeldet.

Wird ferner festgestellt, dass die nächste Meldung länger als die durch den Systemparameter NBRCSCK bzw. NBRCSCKN (siehe Seite 753) festgelegte maximale Wartezeit zur Ausgabe ansteht, reagiert das System entsprechend der DISCON-Angabe in der Verbindungsnachricht (siehe Seite 673):

DISCON=YES: Die Verbindung des betreffenden berechtigten Benutzerprogrammes zur UCON-Task ($CONSOLE) wird abgebrochen.

DISCON=NO: Alle Meldungen, die zur Ausgabe an das berechtigte Benutzerprogramm anstehen, werden gelöscht und durch eine einzelne Meldung `NBR0601` ersetzt.

## 17.1.4 Nachrichtenformate

### Nachrichtenformate und Protokollversionen

Jede Nachricht, die ein berechtigtes Benutzerprogramm sendet, darf höchstens 31 KB lang sein. Längere Nachrichten werden mit `REJ1` abgewiesen. Nachrichten, die das Benutzerprogramm empfangen soll, sind damit maximal 31 KB groß plus die Länge des ggfs. hinzugefügten Headers (siehe Makro NBMHE, Seite 680), denn von anderen Absendern stammende Nachrichten sind sowieso viel kleiner (so beträgt die maximale Länge einer mit MSG7X ausgegebenen Meldung z.B. 4 KB).

Der Text dieser Nachrichten hat grundsätzlich das gleiche Format wie bei Ein-/Ausgaben an physikalischen Konsolen (siehe voriges Kapitel). Zusätzliche Formate ergeben sich ausschließlich, wenn das berechtigte Benutzerprogramm als Server eines Operator-Spezialkommandos fungiert. Diese Erweiterungen sind in ab Seite 684 beschrieben.
Für die automatische Auswertung der empfangenen Meldungen ist ein reines Textformat jedoch ungünstig, da jede Einzelinformationen – z.B. der Wert eines bestimmten Inserts in einer Meldung – in dem Text erst gesucht werden müsste. Das Format von Ausgabetexten ist aber keine garantierte, bei Versionswechsel unverändert bleibende Schnittstelle.
Das System bietet daher an, mit berechtigten Benutzerprogrammen formatierte Nachrichten auszutauschen, die neben dem Text zusätzliche Informationen enthalten. Gesteuert wird dies durch die Protokollversion, die das berechtigte Benutzerprogramm beim Verbindungsaufbau zu $CONSOLE beantragt. Dieser Vorgang ist für Anschlüsse unter generiertem und unter dynamischem Berechtigungsnamen unterschiedlich und kann den entsprechenden Abschnitten auf Seite 668 entnommen werden.
Die Bedeutung der ausgewählten Protokollversion ist jedoch in allen Fällen gleich.

*Protokollversion 0*

Hinter dem Begriff Protokollversion 0 verbirgt sich die einfache Vereinbarung, dass sich der Nachrichtenaustausch zwischen System und berechtigtem Benutzerprogramm in beide Richtungen auf den Nachrichtentext beschränkt. Zusätzliche Informationen werden nicht ausgetauscht.

*Protokollversion 1*

Vereinbart das berechtigte Benutzerprogramm mit der $CONSOLE-Schnittstelle die Protokollversion 1, erwartet das System als Eingabe weiterhin nur Texte der bekannten Formate. Beim Senden von Nachrichten durch das berechtigte Benutzerprogramm bestehen also keinerlei Unterschiede zu Protokollversion 0.
Nachrichten vom System erhält das berechtigte Benutzerprogramm jedoch in erweiterter Form, bestehend aus einem Nachrichtenkopf (Header), dem eigentlichen Text, so wie er bei Protokollversion 0 netto übertragen worden wäre, sowie, falls es sich bei der Nachricht um eine mit MSG7[X] ausgegebene Meldung handelt, ein Füllbyte mit nachfolgender datenorientierten Beschreibung („Mapping-Format“) des Meldungsaufbaus.

Eine DSECT zur Beschreibung des Nachrichtenkopfes kann durch den Makro NBMHE erzeugt werden, der im Anschluss beschrieben ist.
Aufbau und Inhalt der datenorientierten Meldungsbeschreibung sind der Beschreibung des Makros NBMAP auf Seite 683 zu entnehmen.

*Protokollversion 2*

Vereinbart das berechtigte Benutzerprogramm mit der $CONSOLE-Schnittstelle die Protokollversion 2, erhält es alle Nachrichten vom System in erweiterter Form. Bezüglich des Empfangs von Meldungen durch das berechtigte Benutzerprogramm besteht also kein Unterschied zu Protokollversion 1.
Vom berechtigten Benutzerprogramm an das System gesendete Nachrichten müssen hier jedoch die gleiche, erweiterte Form aufweisen; die Übertragung des reinen Nachrichtentextes reicht nicht mehr aus.
Im Regelfall bringt die Erzeugung des Nachrichtenkopfes für das berechtigte Benutzerprogramm lediglich zusätzlichen Aufwand (und Fehlerquellen), aber keinen besonderen Nutzen mit. Notwendig ist die Verwendung der Protokollversion 2 ausschließlich, wenn das berechtigte Benutzerprogramm als Server für ein verwaltbares Operator-Spezialkommando dienen soll. Die dann geltenden Besonderheiten sind ab Seite 684 beschrieben.

### Makro NBMHE

Der Makro NBMHE beschreibt das Format des Nachrichtenheaders. Die Assembler-Schnittstelle hat folgendes Aufrufformat:

| **Makro** | **Operanden** |
|---|---|
| NBMHE | MF=D/C, PREFIX=prefix, MACID=macid, VER=1/2/3 |

Für den Benutzer ist die Angabe VER=1 nicht erlaubt!

Der Header hat folgenden Aufbau:

| gültig für V01, V02 und V03 | | | | | | | Erweiterung für V02 und V03 | | | | | ... für V03 | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| V | DS | AT | LH | L2 | L1 | D | AK | SP | BC | RC2 | RC1 | FL | UID |

Dabei bedeutet:

V Versionsnummer des Makros NBMHE, der diese Struktur generiert; 1 Byte

DS Dialogstatusbyte; 1 Byte
X'80' Anmeldungsdialog läuft
X'40' Verbindung vorläufig akzeptiert
X'10' endgültige Akzeptierung der Anmeldung
X'08' Anmeldung abgewiesen
X'04' Operator-Logon angefordert
X'02' Kennwort angefordert

AT Auftragstyp; 1 Byte
Werte bei Makro V01: % ? / . * ;
zusätzliche Werte ab
Makro V02: + ! & :

LH Länge des Headers; 2 Byte

L2 Distanz zum datenorientierten Teil; 2 Byte
(L2=0000, falls kein datenorientierter Teil (mit NBMAP beschrieben) vorhanden ist)

L1 Länge des textorientierten Teils; 2 Byte

D Ziel einer Meldung; 4 Byte

Erweiterung für NBMHE Version 02 und 03:

AK Auftragskennzeichen abhängig vom Auftragstyp; 4 Byte

SP Indikator für Senderprofil; 1 Byte

BC Berechtigungscodemenge; 5 Byte

RC2 Kommandoreturncode Subcode2 zum Verschicken an bzw. beim Empfang von Kommandogeber; 1 Byte

RC1 Kommandoreturncode Subcode1 zum Verschicken an bzw. beim Empfang von Kommandogeber; 1 Byte

Erweiterung für NBMHE Version 03:

FL Flag-Byte, das den Typ des Kommandogebers ausgibt; 1 Byte

| | |
|---|---|
| X'80' | physikalische Konsole |
| X'40' | logische Konsole |
| X'20' | SCI-Task (siehe Hinweis zu FL, Seite 682) |

UID Benutzerkennung des Kommandogebers; 8 Byte

*Hinweise zu den einzelnen Feldern*

Die meisten Felder des NBMHE sind nur in bestimmten Zusammenhängen relevant, die in der Tabelle aufgeführt sind. Sollte der jeweilige Zusammenhang nicht gegeben sein, so bedeutet dies bei vom System empfangenen Nachrichten einheitlich, dass der Feldinhalt undefiniert ist, und bei an das System gesendeten Nachrichten, dass der Feldinhalt vom System nicht ausgewertet wird, also beliebig gewählt werden kann.
Ferner ist der von der Art des verwendeten Berechtigungsnamens anhängige Anmeldedialog des berechtigten Benutzerprogrammes aus Gründen der Übersichtlichkeit separat beschrieben, siehe Seite 668. Die Tabelle bezieht sich auf den – von der Berechtigungsnamensart unabhängigen – Nachrichtenaustausch nach erfolgreichem Verbindungsaufbau (Dialogstatusbyte = 0).

V Immer relevant
Die Versionsnummer wird durch den Parameter VER des NBMHE-Makros bestimmt. Das System verwendet für seine Nachrichten stets die neueste Version. Das Benutzerprogramm ist jedoch nicht verpflichtet, beim Senden von Nachrichten die gleiche Version zu verwenden; mit Ausnahme von Version 1 akzeptiert das System alle kleineren Versionen kompatibel und eventuelle größere Versionen mit der Einschränkung, dass neu hinzugekommene Felder natürlich nicht berücksichtigt werden können.

DS Immer relevant
Das Dialogstatusbyte zeigt an, in welcher Phase des Anmeldungsdialogs sich das berechtigte Benutzerprogramm befindet. Von Null verschiedene Werte können ausschließlich während des Anmeldedialogs vorkommen.
Nach erfolgreichem Verbindungsaufbau ist das Byte immer 0.

AT Immer relevant
Typkennzeichen der Nachricht (siehe Seite 632).

LH Länge des Headers, gleichzeitig – da der Nachrichtentext unmittelbar hinter dem Header folgt – die Distanz vom Anfang der Nachricht zum Text.
Das Feld muss mit der angegebenen NBMHE-Version übereinstimmen (d.h. 25 Byte für Version 2, 34 Byte für Version 3). Um auch für künftige Versionen korrekt arbeiten zu können, muss die Länge trotzdem immer dem Längenfeld entnommen werden.

L2 (DLL2) Immer relevant
Distanz vom Anfang der Nachricht zum datenorientierten Teil (NBMAP).
Ist kein datenorientierter Teil vorhanden, ist die Distanz 0.
Wichtig: Ist ein datenorientierter Teil vorhanden, darf er nicht unmittelbar hinter dem Nachrichtentext liegen, sondern muss durch mindestens ein Füllbyte von ihm getrennt sein. L2 ist also zwangsläufig größer als LH + L1.

L1 (DLL1) Immer relevant

D Nur relevant bei empfangenen Nachrichten

AK Nur relevant beim Empfang von Kommandos (AT = „ / “) und Kommandozusatzinformationen (AT = „ : “) sowie beim Senden von Nachrichten im Rahmen der Kommandobehandlung (näheres siehe „Operator-Spezialkommandos“, ab Seite 684).

SP Nur relevant beim Senden von Nachrichten
Dieses Byte muss immer auf Null gesetzt werden. Andere Werte sind für Systemanwendungen reserviert.

BC Nur relevant, wenn SP ungleich Null ist (also ausschließlich für Systemanwendungen).

RC2 Nur bei Nachrichten vom Typ „Kommandoende“ (AT = „ ! “) relevant

RC1 Nur bei Nachrichten vom Typ „Kommandoende“ (AT = „ ! “) relevant

FL Nur beim Empfang von Kommandos (AT = „ / “) relevant
Der Begriff „SCI-Task“ bezeichnet hier eine Benutzertask mit dem zur Eingabe des Kommandos notwendigen Privileg (bei selbstdefinierten Operator-Spezialkommandos muss dies nicht das OPERATING-Privileg sein).

UID Nur beim Empfang von Kommandos (AT = „ / “) relevant

i Die Versionsnummer des Headers ist nicht mit der Protokollversionsnummer der Verbindung zu verwechseln.

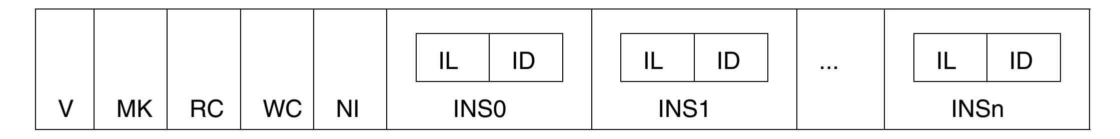
Das Feld LH befindet sich in allen Versionen auf Distanz X'03' im Header. Ihm ist die Distanz zum Beginn des Folgetextes zu entnehmen, da der Folgetext unmittelbar hinter NBMHE beginnt.

## Makro NBMAP

Der Makro **NBMAP** beschreibt den Nachspann der Nachricht. Die Assembler-Schnittstelle hat folgendes Aufrufformat:

| Makro | Operanden |
|---|---|
| `NBMAP` | `MF=D/C/L,PREFIX=chars 1, MACID=macid 1..3, VERSION=1`<br>`,MSGKEY=chars 7..7,MSGRC=chars 1..1`<br>`,MSGWEIGHT=msgweight,MSGINSNUM=integer 0..15`<br>`,INSERTS=((length1,distance1),...,(length,distancex))` |

Die Struktur hat folgenden Aufbau:




V Versionsnummer des Makros NBMAP, der diese Struktur generiert; 1 Byte

MK Meldungsschlüssel; 7 Byte

RC Routing-Code; 4 Byte (linksbündig, restliche Stellen mit 0 aufgefüllt)

WC Gewicht der Meldung; 1 Byte

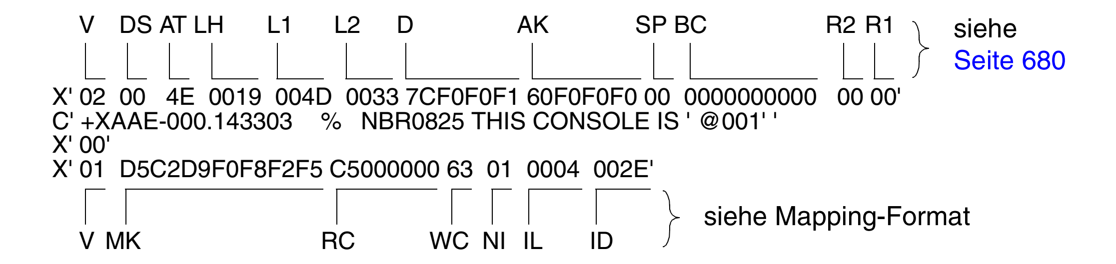
NI Anzahl der Inserts; 1 Byte

IL Länge des aktuellen Inserts; 2 Byte

ID Distanz des aktuellen Inserts ab Anfang des Meldungstextes; 2 Byte

INS0 .... INSn 0...n Inserts der Meldung

### Beispiel mit Nutzung des Nachrichten-Headers V02



## 17.2 Operator-Spezialkommandos in berechtigten Benutzerprogrammen

Ein Spezialkommando ist ein besonderes Operatorkommando, dessen Format und Funktion die Systembetreuung festlegt.
Das System leitet ein empfangenes Spezialkommando einem berechtigten Benutzerprogramm zu, das das Kommando auswertet und die gewünschten Tätigkeiten ausführt. Die Bearbeitung von Spezialkommandos ist somit eine spezielle Anwendung von berechtigten Benutzerprogrammen.

Es gilt:

- Für jedes Spezialkommando ist eine Operationsbezeichnung (= Name) festzulegen. Diese darf nicht mit den Operationsbezeichnungen der gewöhnlichen Operatorkommandos übereinstimmen. Die Operationsbezeichnung „CANCEL“ ist z.B. als Operationsbezeichnung für ein Spezialkommando unzulässig.
- Jedes Spezialkommando ist einem Aufgabengebiet zuzuteilen (das Kommando wird mit einem Berechtigungsschlüssel versehen). Falls ein Spezialkommando sich keinem der bestehenden Aufgabengebiete zuordnen lässt, ist es einem privaten Aufgabengebiet zuzuteilen (Berechtigungsschlüssel: W, X, Y, Z).
- Um die Eingabe eines Spezialkommandos auch aus einer Benutzertask mit Privileg OPERATING zu ermöglichen, ist es in einer SDF-Syntaxdatei mit SDF-A zu definieren. Diese SDF-Syntaxdatei muss der Benutzertask vorgeschaltet werden.
  Mit ADD-CMD ...,IMPLEMENTOR=APPLICATION wird festgelegt, dass die Kommandobearbeitung einem berechtigten Benutzerprogramm folgt.

So ist beispielsweise das Kommando DADM zur Administration des Datenkommunikationssystems ein Spezialkommando, das durch das Programm TDADM unter dem Berechtigungsnamen DADM der Systembedienung (Aufgabengebiet C) zur Verfügung gestellt wird.

Für das Verhalten von Operator-Spezialkommandos aus der Sicht des Benutzers ist maßgebend, ob die Kommandos die Eigenschaft der Verwaltbarkeit besitzen.
Man unterscheidet verwaltbare und nicht verwaltbare Operator-Spezialkommandos.

### Verwaltbare Spezialkommandos

Sie werden vom System genauso verwaltet wie gewöhnliche Operator-Kommandos.
Der Austausch von kommandogebundenen Nachrichten ist möglich.
Die Kommando-Endemeldung wird – außer in Fehlerfällen – von der kommandoausführenden $CONSOLE-Anwendung veranlasst und enthält einen auswertbaren Returncode.

### Nicht-verwaltbare Spezialkommandos

Sie werden vom System als „beendet“ angesehen, sobald sie (mit DCAM) erfolgreich an die kommandoausführende $CONSOLE-Anwendung abgeschickt wurden. Der Austausch von kommandogebundenen Meldungen ist sonst nicht möglich. Verwendet der Operator ein nicht-verwaltbares Spezialkommando, so erhält er die Meldung

```
NBR0740 COMMAND TERMINATED '<cmd-name>':
        RESULT: (SC=00, SC1=00,MC=NBR0768); DATE:<date>
```

Das bedeutet jedoch nur, dass das Kommando an den zuständigen Server weitergegeben wurde. Das endgültige Kommandoergebnis findet sich in keiner Abschlussmeldung und Kommandoauftragskennzeichen finden sich nicht als Meldungsauftragskennzeichen in den Kommandoergebnissen wieder.

| **Kommando** | **Bedeutung** |
|---|---|
| CONNECT-CMD-SERVER | Verknüpfung von berechtigten Anwendungen mit Operatorkommandos herstellen |
| DISCONNECT-CMD-SERVER | Verknüpfung von berechtigten Anwendungen mit Operatorkommandos lösen |
| SHOW-CMD-ATTRIBUTES | Informationen über Operatorkommandos ausgeben |

Tabelle 51: Kommandoübersicht Operator-Spezialkommandos

## 17.2.1 Kommandodefinition

Alle Eingaben, die an einer Konsole gemacht werden, gehen zunächst an UCON, die zentrale Konsoltask des Systems. Handelt es sich bei der Eingabe um ein Kommando, muss UCON zunächst die an den Routing-Code des Kommandos geknüpfte Berechtigungsprüfung durchführen. Erst wenn die bestanden ist, darf das Kommando an den zuständigen Kommandoserver weitergeleitet werden.
Da der Routing-Code eines Kommandos (und bei Operator-Spezialkommandos auch der Kommandoserver) jedoch nicht Bestandteil der SDF-Syntaxdatei ist, muss sich UCON eine eigene Kommandotabelle halten, in der die benötigten Informationen enthalten sind. Kommandos, die in der UCON-Kommandotabelle nicht enthalten sind, können an Konsolen nicht ausgeführt werden. Operator-Spezialkommandos müssen daher als Erstes in diese Tabelle eingetragen werden.

### Statische Einträge

Eine Möglichkeit zur Eintragung von Operator-Spezialkommandos in die UCON-Kommandotabelle ist die Parameterservice-Anweisung ADD-CMD-ENTRY (siehe Seite 107). Die dadurch erzeugten Einträge sind statisch, d.h. bleiben für die gesamte Dauer der Session erhalten. Als zuständiger Kommandoserver kann daher nur eine $CONSOLE-Anwendung mit (statisch) generiertem Berechtigungsnamen angegeben werden (ist der angegebene Berechtigungsname noch nicht bekannt, wird er automatisch generiert).
Ferner besteht hier keine Möglichkeit, das Kommando als verwaltbar zu deklarieren.

Soll ein Operator-Spezialkommando verwaltbar ablaufen oder von einer $CONSOLE-Applikation mit dynamischen Berechtigungsnamen ausgeführt werden, muss der Eintrag in die Kommandotabelle mit Hilfe des Kommandos CONNECT-CMD-SERVER dynamisch erfolgen.

### Dynamische Einträge

Eine $CONSOLE-Anwendung, die als Server für ein Operator-Spezialkommando fungieren soll, kann sich selbst mit dem Kommando CONNECT-CMD-SERVER dynamisch bei UCON als Kommandoserver anmelden. UCON erzeugt dann automatisch einen entsprechenden Eintrag in der Kommandotabelle.
Zu beachten ist dabei, dass pro Kommando bis zu vier Einträge in der Kommandotabelle erlaubt sind. CONNECT-CMD-SERVER kann also auch für bereits eingetragene Kommandos (auch Systemkommandos) abgesetzt werden (wobei allerdings nicht alle Eigenschaften des Kommandos geändert werden können, siehe Kommandobeschreibung). Der neue Eintrag „überlagert“ dann den alten, und die Kommandos werden fortan der $CONSOLE-Anwendung zugestellt, die den neuesten Eintrag besitzt. Welche Einträge alles existieren, lässt sich mit dem Kommando SHOW-CMD-ATTRIBUTES anzeigen.

Ein dynamischer Eintrag in der UCON-Kommandotabelle wird automatisch wieder entfernt, wenn die Verbindung von $CONSOLE zum zuständigen Kommandoserver abgebaut wird. Ferner kann die Anwendung ihn mit Hilfe des Kommandos DISCONNECT-CMD-SERVER explizit löschen.
Ist der entfernte Eintrag der einzige für dieses Kommando gewesen, ist das Kommando anschließend nicht mehr bekannt. Gibt es weitere Einträge, so wird (bzw. bleibt) der neueste der noch existierenden Einträge wirksam; der Eigentümer dieses Eintrags bekommt die Kommandos zugestellt.

### Definition in der Syntaxdatei

SDF-A bietet die Möglichkeit, Syntaxdatei-Einträge für Operator-Spezialkommandos zu erstellen (Anweisung ADD-CMD <name>, IMPLEMENTOR= *APPLICATION). Ein solcher Syntaxdatei-Eintrag ist notwendig, wenn das Kommando Kennwörter enthält, die nicht in der CONSLOG protokolliert werden sollen. Ein solches Kommando ist mit PASSWORD-POSSIBLE=*YES in die UCON-Kommandotabelle einzutragen. Die CLOG-Task ruft dann vor der Protokollierung in die CONSLOG SDF auf und lässt alle als „geheim" (SDF-A: ADD-OPERAND <name>, SECRET-PROMPT=*YES) gekennzeichneten Operanden entfernen. Existiert kein Syntaxdatei-Eintrag, wird das Kommando unverändert im Klartext protokolliert.
Darüber hinaus können Operator-Spezialkommandos, für die ein Syntaxdatei-Eintrag vorhanden ist, nicht nur an Konsolen verwendet werden, sondern in allen Benutzertasks, die gemäß den in der Syntaxdatei eingetragenen Kommandoprivilegien dazu berechtigt sind. Eine Berechtigungsprüfung anhand des Routing-Codes findet bei Eingabe in einer Benutzertask nicht statt.

Soll ein Operator-Spezialkommando durch Eintrag in die Systemsyntaxdatei einem bestimmten Benutzerkreis zur Verfügung gestellt werden, ist folgende Besonderheit zu beachten:
Der Syntaxdatei-Eintrag ersetzt den Eintrag in die UCON-Kommandotabelle nicht. Kommandos, die zwar in der Systemsyntaxdatei, aber nicht in der UCON-Kommandotabelle eingetragen sind, werden mit der Meldung NBR0744 zurückgewiesen. Ebenso wird mit zusätzlichen, UCON unbekannten Alias-Namen eines eigentlich bekannten Kommandos verfahren.

Das Privileg OPERATING kann an beliebige Benutzerkennungen vergeben werden. Damit verbunden ist die Erlaubnis, fast alle sog. Operator-Kommandos nutzen zu können.

## 17.2.2 Nachrichtenformate

Folgende Formate werden beschrieben:

- Formate der Kommandos beim Empfangen von Operatorkommandos
- Formate beim Senden von Kommandoergebnissen und Kommando-Zusatzinformationen
- Formate beim Senden von Kommandoende-Nachrichten
- Formate beim Empfang von Kommando-Zusatzinformationen

### Formate der Kommandos beim Empfangen von Operatorkommandos

```
              ⎧ tsn  ⎫
[header]  /  ⎨ an    ⎬ :␣␣ cmd ␣[operands]
              ⎩ (mn) ⎭
  |         |   |    |  |     |
  n      +  0   1    4  5     8
```

header Bei Anwendungen, die sich an $CONSOLE mit der Protokollversion 01 oder 02 konnektiert haben, enthält das Format einen n Zeichen langen Header, der durch den Makro NBMHE beschrieben wird.

/ Nachrichtentyp-Kennzeichen (Schrägstrich)

tsn Auftragsnummer der Benutzertask, von der das Operator-Spezialkommando eingegeben wurde

an Berechtigungsname des berechtigten Benutzerprogramms, von dem das Kommando eingegeben wurde (4 Zeichen)

mn Mnemotechnischer Gerätename der Bedienstation, an der das Kommando eingegeben wurde (2 Zeichen)

: Doppelpunkt als Trennzeichen

␣ Leerzeichen

cmd Bezeichnung eines Kommandos

operands Operanden des Kommandos



Die Auftragsidentifikation, die ein berechtigtes Benutzerprogramm beim Empfangen des Kommandos im Header mitbekommt, ist eine vom System vergebene, eindeutige Auftragsidentifikation. Sie ist nicht identisch mit dem bei der Kommadoeingabe vergebenen Auftragskennzeichen.
Die interne Auftragsidentifikation ist für das berechtigte Benutzerprogramm transparent. Das System verwaltet das Kommando anhand dieser Auftragsidentifikation und sorgt dafür, dass die in Zusammenhang mit dem Kommando gesendeten Kommandoergebnisse, die Zusatzanforderungen und die Kommandoendemeldung stets mit dem vom Kommadogeber eingegebenen Auftragskennzeichen versorgt werden und den Kommandogeber erreichen.
Vorrausetzung ist, dass im Header die interne Auftragsidentifikation und der Auftragstyp (Kommandoergebnis (+), Kommando-Zusatzinformation-Anforderung (&) oder Kommandobeendigung (!)) richtig versorgt sind.

*Beispiele*

1. ohne Header

```
/OPR1:  SPECI OP1=VALUE1,OP2=VALUE2
```

2. mit Header

```
<header>/OPR1:  SPECI OP1=VALUE1,OP2=VALUE2
```

Zur Datenstruktur des Headers siehe Makro NBMHE (Seite 680).

### Formate beim Senden von Kommandoergebnissen und Kommando-Zusatzinformationen

```
<header> text [<mapping-format>]
```

header
: Im Header muss der entsprechende Auftragstyp gekennzeichnet und das Feld Auftragsidentifikation mit der aus dem Header des Kommandos entnommenen Auftragsidentifikation versorgt werden. Das System sorgt dafür, dass der Text an der Kommandoeingabestelle mit dem im Kommando enthaltenen Auftragskennzeichen erscheint.

text
: formatfreier (beliebiger) Text oder Nachrichtentext im MSG7X-Format

mapping-format
: Sendet eine $-CONSOLE-Anwendung Nachrichten des Typs „+“ oder „&“ unter Angabe eines Mapping-Formats (siehe Makro NBMAP, Seite 683), und war der Absender des Kommandos eine Benutzertask, so ignoriert das System für die Meldungsausgabe den angegebenen Nachrichtentext.

Allerdings wird der angegebene Nachrichtentext zur Aufnahme in die CONSLOG herangezogen. Er sollte daher MSG7X-Format haben. Desweiteren werden systeminterne Prüfungen (z.B. die Meldungsnummer an Distanz 4 im Nachrichtentext) durchgeführt, die i.d.R. nur bei einem MSG7X-Format erfolgreich sein können.

Der Absender des Kommandos erhält als Nachricht die Ausgaben, die der Makro MSG7X mit den im Mapping-Format angegebenen Parametern erzeugt. Dabei wird automatisch die in der Umgebung des Kommandosenders gültige Sprache verwendet.
Der Zweck dieses MSG7X-Aufrufes ist es also, die sprachlich erwünschte Nachrichtentext-Ausgabe zu erreichen.

### Formate beim Senden von Kommandoende-Nachrichten

```
header maincode
```

header — Im Header muss der Auftragstyp „!“ eingetragen und die Felder RC1 und RC2 mit dem Subcode1 bzw. dem Subcode2 des Kommandoergebnisses versorgt werden.
Ferner muss das Feld Auftragsidentifikation mit der aus dem Header des Kommandos entnommenen Auftragsidentifikation versorgt werden. Das System sorgt dafür, dass die Meldung `NBR0740` an der Kommandoeingabestelle ausgegeben wird.
`NBR0740 COMMAND COMPLETED 'xxxx'; (RESULT:....); DATE:.....`

Die Meldung enthält auch das bei der Eingabe des Kommandos eingegebene Auftragskennzeichen und ist mit einem Ausrufezeichen gekennzeichnet.

maincode — 7 Zeichen langer Maincode des Kommandoergebnisses



Nach dem Verschicken einer Kommandobeendigungsanzeige darf ein berechtigtes Benutzerprogramm die interne Auftragsidentifikation nicht mehr weiter verwenden. Aufträge für Kommandos, deren Beendigung bereits angezeigt wurde, oder Aufträge mit falscher Auftragsidentifikation werden vom System mit der Meldung `REJ11` abgewiesen.

### Formate beim Empfang von Kommando-Zusatzinformationen

```
          ⎧ tsn  ⎫
header  : ⎨ an   ⎬  [-mid]:[text]
          ⎩ (mn) ⎭
```

header n Zeichen langer Header, der durch den Makro NBMHE beschrieben wird

: Nachrichtentyp-Kennzeichen (Doppelpunkt)

an Berechtigungsname des antwortenden berechtigten Benutzerprogramms (4 Zeichen)

mn Mnemotechnischer Gerätename der antwortenden Bedienstation (2 Zeichen)

tsn Auftragsnummer der antwortenden Benutzertask

- Bindestrich

mid Meldungsauftragskennzeichen (A..Z, 0-9, @, #, $); wird bis zu 3 Zeichen mit führenden Nullen ergänzt. Standardwert = 000

: Doppelpunkt als Trennzeichen

text beliebiger Antworttext

## 17.3 Software-Produkte OMNIS, PROP-XT und OMNIS-PROP

Das Software-Produkt **OMNIS** ist ein Steuersystem zur zentralen Bedienung mehrerer Anwendungen in BS2000/OSD.

In BCAM kann eine Datenstation prinzipiell zu einem Zeitpunkt nur eine einzige Verbindung zu einem Kommunikationspartner unterhalten. OMNIS ist ein Programm, das es ermöglicht, diese Einschränkung zu umgehen.

OMNIS ermöglicht die gleichzeitige Verbindungsaufnahme von einer Datenstation zu mehreren Partnern in einem Rechnerverbund. Über OMNIS kann man mit einem Partner auch mehrfach in Verbindung treten. Außerdem können Sie mit OMNIS auch 9750-Datenstationen für Anwendungen verwenden, die Datenstationen eigentlich gar nicht unterstützen, wie z.B. UCON.

OMNIS ist im Handbuch „OMNIS/OMNIS-MENU“ [34] ausführlich beschrieben.

**PROP-XT** ist ein Produkt im Rahmen der Automatisierung im Data Center für das Aufgabengebiet der Server-Steuerung.
Es ermöglicht programmiertes Operating durch vom Benutzer mit den komfortablen Sprachmitteln von SDF-P erstellte Administrationsprozeduren.

PROP-XT ist im Handbuch „PROP-XT“ [42] ausführlich beschrieben.

Neben PROP-XT existiert auch noch das ältere, auf Assembler- bzw. Makrosprache basierende Produkt OMNIS-PROP. Es hat eine weniger komfortable Benutzeroberfläche und setzt Assembler-Programmierkenntnisse voraus.

Das Produkt **OMNIS-PROP** ist eine spezielle Exit-Routine des Produkts OMNIS.
OMNIS-PROP ermöglicht es, definierbare Uhrzeiten, Zeitintervalle und Systemmeldungen als Ereignisse zu deklarieren, die bei ihrem Eintreffen Administrationsprozeduren aktivieren, welche Operatoraufgaben erledigen (z.B. Eingabe von Operatorkommandos).
Diese Administrationsprozeduren sind entweder bereits vorhanden oder vom Benutzer in Assembler zu programmieren, wobei er die Makros von OMNIS-PROP verwendet.

OMNIS-PROP ist im Handbuch „OMNIS-PROP“ [36] ausführlich beschrieben.

## 17.4 Kommandodateien für den Operator

### Kommandodateien beim Systemstart

Nach abgeschlossener Systemeinleitung hat der Operator gewöhnlich den Dialogbetrieb vorzubereiten. Er nimmt unter anderem das Datenkommunikationssystem in Betrieb und leitet die Teilhaberanwendungen ein.



Das Starten des Datenkommunikationssystems kann bereits vor „System Ready“ erfolgen. Zu diesem Zweck können in der Startup-Parameterdatei (Abschnitt BCAM) die entsprechenden BCAM-Kommandos eingetragen werden. Siehe dazu auch das Handbuch „BCAM“ [4].

Die Kommandos, die zur Vorbereitung des Dialogbetriebs erforderlich sind, lassen sich in einer Kommandodatei hinterlegen.
Wenn der Name dieser Kommandodatei bereits im Startup-Parameterservice (Systemparameter CMDFILE) festgelegt wurde, so bearbeitet das System die Datei automatisch unmittelbar nach dem Systemstart.
Der Operator kann entweder durch Änderung des Systemparameters CMDFILE mittels Parameterservice oder beim DIALOG-Startup eine andere Kommandodatei ausführen lassen oder manuell Kommandos eingeben (siehe auch Systemparameter NBRUNUID).

### Kommandodateien bei häufig vorkommenden Kommandofolgen

Kommandofolgen, die der Operator wiederholt einzugeben hat – beispielsweise bei häufig wiederkehrenden Aufträgen – lassen sich in Kommandodateien speichern.

Der Operator startet die Ausführung einer Kommandodatei mit einem RUN-Kommando. Dies kann entweder von der Bedienstation aus erfolgen, oder das RUN-Kommando wird von einem berechtigten Benutzerprogramm abgesetzt. In beiden Fällen muss darauf geachtet werden, dass der aufrufenden Bedienstation oder dem berechtigten Benutzerprogramm alle Aufgabengebiete zugeordnet sind, auf die sich die Kommandos in der Kommandodatei beziehen. Neben den Kommandodateien kann die Systembetreuung ENTER-Jobs aufbauen, die der Operator von der Bedienstation startet. Da Kommandofolgen in ENTER-Jobs Benutzerkommandos enthalten und nach dem Start keinerlei besondere Maßnahmen seitens des Operators erfordern, wird im Folgenden nicht weiter auf diese Kommandofolgen eingegangen.

Das Anbieten der Operator-Funktionalität in beliebigen Benutzertasks mit dem Privileg OPERATING ermöglicht auch das Bereitstellen von beliebigen Prozeduren (mit und ohne SDF-P) für den Operator. Diese Kommandodateien können eine beliebige Mischung aus Operator- und Benutzerkommandos enthalten. Das Starten der Prozeduren mit CALL-PROCEDURE und ENTER-PROCEDURE ist jedoch nur aus Benutzertasks möglich und nicht von Bedienstationen oder berechtigten Anwendungen. Die Funktionalität entspricht in vollem Umfang den Prozeduren in Benutzertasks und wird hier nicht beschrieben.

Die nachstehenden Ausführungen beziehen sich also ausschließlich auf Operator-Kommandofolgen.

| **Kommando** | **Bedeutung** |
|---|---|
| AGOGO | Kommandodatei fortsetzen |
| ASTOP | Bearbeitung der Kommandodatei anhalten und die Bedingung zum Fortsetzen festlegen |
| CANCEL-RUN-PROCESS | Abbrechen einer Kommandodatei |
| ENTER-JOB | Starten eines ENTER-Jobs |
| RUN | Kommandodatei starten |
| SHOW-PENDING-MSG | Wiederholtes Anzeigen von Systemmeldungen |

Tabelle 52: Kommandoübersicht zu Kommandodateien

### 17.4.1 Ausführen und Abbrechen einer Kommandodatei

#### Ausführen einer Kommandodatei

Eine Kommandodatei lässt sich von jeder Bedienstation und von jedem berechtigten Benutzerprogramm aufrufen, sofern sie den Berechtigungsschlüssel E besitzen (siehe Abschnitt „Berechtigte Benutzerprogramme mit Operatorfunktionen“ auf Seite 663).
Der aufrufenden Bedienstation bzw. dem Benutzerprogramm müssen jedoch alle Aufgabengebiete zugeordnet sein, auf die sich die Kommandos in der Kommandodatei beziehen (über den Berechtigungsschlüssel).

Das System bearbeitet zu einem gegebenen Zeitpunkt höchstens eine Kommandodatei. Empfängt es während der Bearbeitung einer Kommandodatei ein weiteres RUN-Kommando, so verzögert es die Ausführung dieses Kommandos so lange, bis die aktuelle Kommandodatei abgearbeitet ist.

Dabei geht das System wie folgt vor:

- Stehen mehrere RUN-Kommandos in einer Kommandofolge, so werden sie in der Reihenfolge bearbeitet, wie sie in der Kommandofolge stehen.
- Wird eine RUN-Prozedur mit dem Kommando CANCEL-RUN-PROCESS gestoppt, so werden auch alle in der RUN-Datei enthaltenen RUN-Kommandos nicht mehr ausgeführt.

Wenn die Ausführung einer Kommandodatei gestartet wird, zeigt dies Meldung `NBR1000` an. Der Aufrufer einer Kommandodatei erhält nur Meldungen, die beim Ausführen der einzelnen Kommandos entstehen; die Kommandos selbst werden ihm i.d.R. nicht angezeigt. Will der Kommandogeber das Kommando aus der RUN-Datei auf der Konsole mitprotokollieren, kann er dies mit dem Systemparameter NBRUNSP=Y erreichen. Unabhängig vom Systemparameter NBRUNSP werden Kommandos aus RUN-Dateien in der Protokolldatei (CONSLOG) festgehalten.

Enthält eine Kommandodatei Spezialkommandos, so läuft sie nur dann korrekt ab, wenn das berechtigte Benutzerprogramm bereits gestartet worden ist, das die Spezialkommandos empfangen und ausführen soll.

Der Ablauf einer Kommandodatei, die Spezialkommandos enthält, muss daher mit dem Start des berechtigten Benutzerprogramms koordiniert werden.
Dazu sind folgende Aktionen von der RUN-Prozedur aus zu veranlassen:

1. berechtigte Benutzerprogramme zur Bearbeitung von Spezialkommandos starten (mit ENTER-JOB-Kommandos)
2. Bearbeitung der Kommandodatei anhalten und die Bedingung zum Fortsetzen festlegen (mit einem ASTOP-Kommando)

Um die Anlaufzeit auszunutzen, die die Tasks benötigen, stellt man die ENTER-JOB-Kommandos am günstigsten an den Anfang der Kommandodatei und das ASTOP-Kommando unmittelbar vor das erste Spezialkommando.

Die Bearbeitung der Kommandodatei wird fortgesetzt, wenn eine der folgenden Bedingungen erfüllt ist:

- Es sind so viele AGOGO-Komandos eingetroffen, wie im ASTOP-Kommando als Bedingung festgelegt ist.
- Die Wartezeit ist verstrichen, ohne dass genügend viele AGOGO-Kommandos eingetroffen sind.

Der Zeitraum, für den die Kommandobearbeitung bei einem ASTOP-Kommando unterbrochen wird, kann mit dem Systemparameter NBRUNWT festgelegt werden.
Er liegt zwischen 10 und 255 Sekunden und beträgt standardmäßig 180 Sekunden.

*Beispiel für Spezialkommando*

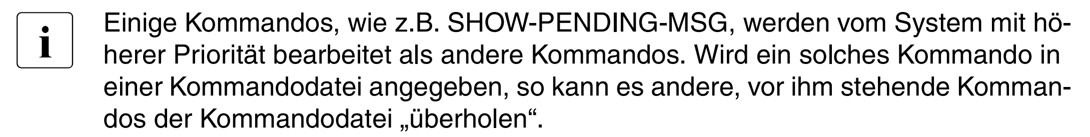
Die Systembetreuung hat für das von ihr betreute System das Spezialkommando SPECI vorgesehen, welches in der Kommandodatei nach dem Systemstart benutzt werden soll.

Diese Datei muss unter anderem folgende Kommandos enthalten:

```
/ENTER-JOB SPEC.CMD-PROC
  :
/ASTOP

/SPECI
   :
```

Die Datei SPEC.CMD-PROC muss dabei eine geeignete Kommandofolge zum Starten des Benutzerprogramms enthalten, das für die Verarbeitung des SPECI-Kommandos erstellt wurde. Die Datei sollte zudem das Kommando AGOGO enthalten, das zusammen mit dem Kommando ASTOP der RUN-Prozedur die Synchronisierung herstellt.


Einige Kommandos, wie z.B. SHOW-PENDING-MSG, werden vom System mit höherer Priorität bearbeitet als andere Kommandos. Wird ein solches Kommando in einer Kommandodatei angegeben, so kann es andere, vor ihm stehende Kommandos der Kommandodatei „überholen“.

Das Zusammenspiel zwischen der Bearbeitung einer Kommandodatei und den Aktionen eines einzelnen berechtigten Benutzerprogramms zeigt das folgende Bild.

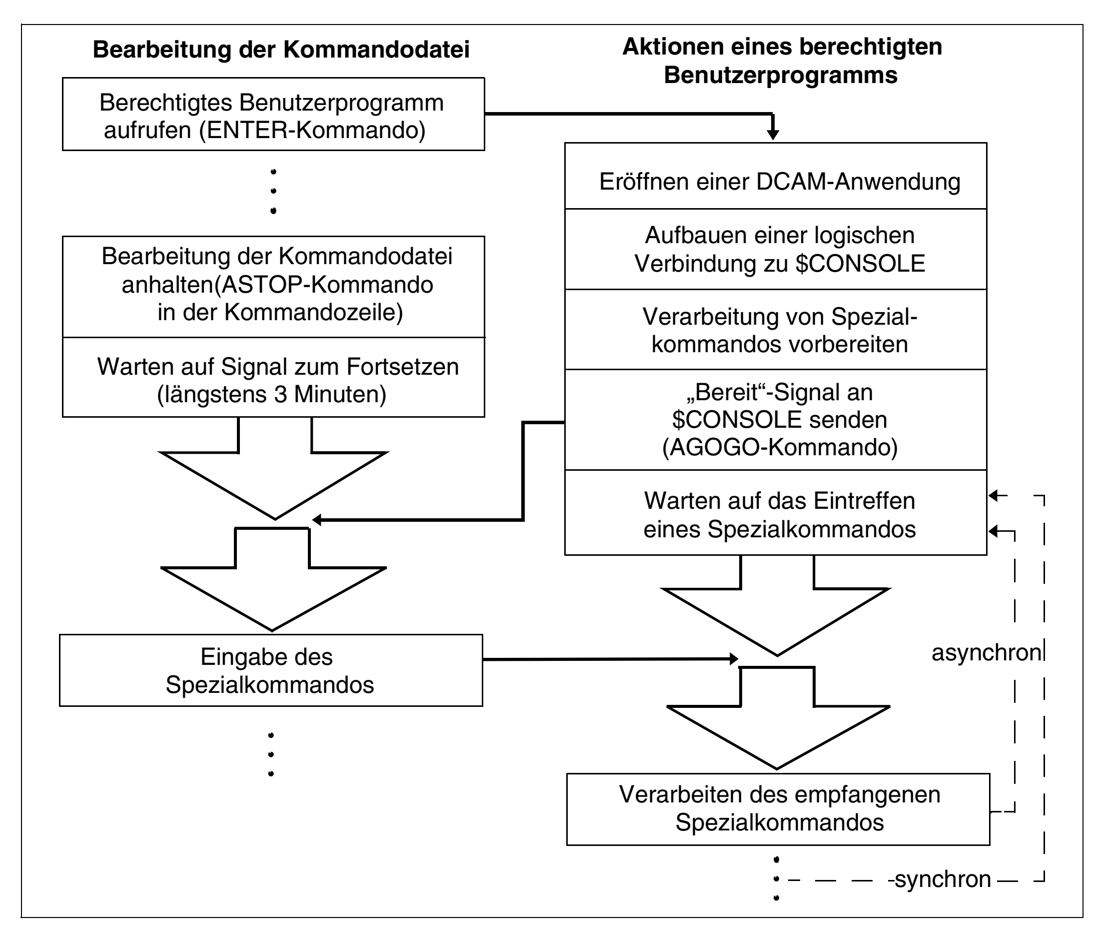



Bild 37: Bearbeiten einer Kommandodatei

### Abbrechen einer Kommandodatei

Das Kommando CANCEL-RUN-PROCESS bricht die Bearbeitung einer gestarteten Kommandodatei ab. Zu seiner Ausführung muss der Operator eine RUN-ID angeben, die jeder gestarteten Kommandodatei nach dem Kommando RUN zugeteilt wird. Kommandodateien, die rekursiv aus einer anderen RUN-Sequenz angestartet werden, erhalten dieselbe RUN-ID wie die erzeugende Kommandodatei. Im Falle des Abbruchs einer Kommandodatei mit CANCEL-RUN-PROCESS werden automatisch auch alle erzeugten Kommandodateien abgebrochen. Das Kommando muss nicht notwendigerweise von derselben Bedienstation eingegeben werden, von dem aus das Kommando RUN gestartet wurde.
Es setzt allerdings den Berechtigungsschlüssel E voraus.

## 17.4.2 Aufbau von Kommandodateien

Eine Kommandodatei, die vom Operator an der Bedienstation gestartet wird, ist eine SAM- oder ISAM-Datei bzw. ein Bibliothekselement (Typ J) auf gemeinschaftlichen Datenträgern. Die Datensätze dürfen beliebige Operatorkommandos enthalten, unter bestimmten Voraussetzungen auch privilegierte und Benutzerkommandos. Die RUN-Kommandofolge darf kein SET-LOGON-PARAMETERS- oder EXIT-JOB-Kommando enthalten. Die Datensätze der Kommandodatei haben entweder feste oder variable Satzlänge, wobei die maximale Satzlänge allerdings 201 Byte beträgt. Es werden keine Fortsetzungszeilen unterstützt.
Die Systembetreuung kann die Kommandodatei durch die Vergabe eines Ausführungskennwortes gegen unberechtigte Ausführung durch den Operator schützen. Die Kommandodatei wird vom System nur dann verarbeitet, wenn der Operator im RUN-Kommando das passende Kennwort angibt. Das mögliche Kennwort wird in der CONSLOG-Datei nicht protokolliert.

RUN-Kommandofolgen können beliebig viele Kommandos aufnehmen. Die Systembetreuung sollte beim Aufbau der Kommandofolge nach spätestens 30 Kommandos das Kommando ASTOP einfügen, um einen strukturierten Ablauf der Kommandos zu gewährleisten. Eine Unterbrechung ist auch dann ratsam, wenn die Ausführung eines Kommandos die vollständige Abarbeitung des vorhergehenden bzw. den Aufruf abhängiger Funktionen bedingt.
Stehen beispielsweise 10 Kommandos in der RUN-Kommandodatei, werden diese Kommandos hintereinander eingelesen und bearbeitet. Wird in der Kommandofolge das Kommando SHOW-PENDING-MSG ausgeführt, sollte die Prozedur nach diesem Kommando mit einem ASTOP-Kommando unterbrochen werden. Der Operator kann daraufhin evtl. ausstehende Fragen beantworten. Nach Ablauf der Wartezeit oder einem AGOGO wird die Kommandofolge weiter bearbeitet.

Ein Vorteil einer mit ASTOP-Kommandos gegliederten RUN-Datei ist, dass die Kommandofolge während der Ausführung nicht unnötig Klasse-4-Speicher belegt.
Das System liest eine so aufgebaute und umfangreiche Kommandodatei abschnittsweise in den Klasse-4-Speicher, d.h. von einem ASTOP-Kommando bis zum folgenden, und führt jeweils den eingelesenen Dateiabschnitt aus, bevor es einen neuen einliest.

Sind DSSM-Kommandos Bestandteile der Kommandodatei, sollte die Systembetreuung Folgendes beachten:
Eine eigene Task arbeitet die DSSM-Kommandos ab. Kommandos, die erst eingegeben werden dürfen, wenn das jeweilige Subsystem geladen ist, müssen warten, bis die DSSM-Kommandos abgearbeitet sind. Entsprechend ist auch an dieser Stelle das Kommando ASTOP in die Kommandodatei aufzunehmen.

Die Kommandodatei CMDFILE (Bibliothekselemente sind nicht zulässig) kann eine Folge von Operatorkommandos enthalten, die bei jedem Systemstart unverändert ablaufen. Sie enthält z.B. Aufrufe für die folgende Kommandofolge:

DCSOF Kommandodatei (Start Option File) zur Inbetriebnahme des Datenkommunikationssystems (DCS)

*Beispiel für eine CMDFILE*

```
/MODIFY-JOB-CLASS CLASS-NAME=JCDIALOG,C-L=0 ——————————————————————————————  (1)
/MODIFY-JOB-CLASS CLASS-NAME=TSOSDIA,C-L=0 ———————————————————————————————  (2)
/MODIFY-JOB-CLASS CLASS-NAME=JCBATCH,C-L=40 ——————————————————————————————  (3)
/SET-MSG-SUPPRESSION MSG-ID=(BLS0980,JMS0154),CONSOLE=C0 —————————————————  (4)
/DCSTART DCSOF=SOF.ACT.DCM8 ——————————————————————————————————————————————  (5)
/ASTOP ———————————————————————————————————————————————————————————————————  (6)
/MOD-TASK-CAT CATEG-NAME=DIALOG,WEIGHT-CODE=400,MIN-ACT=35,MAX-ACT=50 —    (7)
/MOD-TASK-CAT CATEG-NAME=BATCH,WEIGHT-CODE=10,MIN-ACT=5,MAX-ACT=20 ———     (8)
/MOD-TASK-CAT CATEG-NAME=TP,WEIGHT-CODE=500,MIN-ACT=20,MAX-ACT=40 ————     (9)
/IMPORT-PUBSET PUBSET=L ——————————————————————————————————————————————— (10)
/ATTACH-DEVICE (X3,X4,X5,S1,S3,K7,X0,Y3) ——————————————————————————————— (11)
/SET-DISK-P V=(PRI009,PRI010,PRI037,PRI038,SESAMA),USER=SHARE,ASS=OPER   (12)
/SET-DISK-P V=(WORK11,SLED),USER=SHARE,ASS=OPER ———————————————————————— (13)
/START-SUBSYS SUBSYS-NAME=MSCF,SUBSYS-PARAM='FILE=$TSOS.RFA.MSCFSTART'   (14)
/START-SUBSYS SUBSYS-NAME=SPOOL,SUBSYS-PARAM='RSO=STD' ————————————————— (15)
/ASTOP —————————————————————————————————————————————————————————————————— (16)
/MODIFY-JOB-CLASS CLASS-NAME=JCDIALOG,CLASS-LIMIT=20 ——————————————————— (17)
/MODIFY-JOB-CLASS CLASS-NAME=TSOSDIA,CLASS-LIMIT=4 ————————————————————— (18)
```

(1) Keine Jobs für die Jobklasse JCDIALOG zulassen.

(2) Keine Jobs für die Jobklasse TSOSDIA zulassen.

(3) Maximal 40 Jobs für die Jobklasse JCBATCH zulassen.

(4) Meldungen mit den Meldungsnummern `BLS0980` oder `JMS0154` werden an der Bedienstation C0 unterdrückt.

(5) Laden des Datenkommunikationssystems.

(6) Kommandodatei anhalten, bis das Datenkommunikationssystem geladen ist.

(7) Anzahl und Dringlichkeit von Tasks für die Kategorie DIALOG festlegen.

(8) Anzahl und Dringlichkeit von Tasks für die Kategorie BATCH festlegen.

(9) Anzahl und Dringlichkeit von Tasks für die Kategorie TP festlegen.

(10) Pubset L importieren.

(11) Geräte dem System zuschalten.

(12) Vorgaben setzen für die Überwachung von Platten.

(13) Vorgaben setzen für die Überwachung von Platten.

(14) MSCF-Kommunikation aktivieren.

(15) Laden des Subsystems SPOOL mit automatischem Laden von RSO.

(16) Kommandodatei anhalten, bis SPOOL geladen ist.

(17) Maximal 20 Jobs für die Jobklasse JCDIALOG zulassen.

(18) Maximal 4 Jobs für die Jobklasse TSOSDIA zulassen.

Mit dem Systemparameter NBCONOPI=Y kann ein System beantragt werden, bei dem auch bei physikalischen Konsolen ein Logon nötig ist.
In einem solchen System ist also z.B. die Eingabe eines RUN-Kommandos an einer physikalischen Konsole nur nach einem erfolgreichen SET-LOGON-PARAMETERS und einem entsprechenden REQUEST-OPERATOR-ROLE möglich.

Wird bei Verwendung des Systemparameters NBCONOPI=Y der Dateiname der RUN-Datei ohne Benutzerkennung angegeben, wird die Datei zuerst unter der Kennung des Operators, dann unter der im Systemparameter NBRUNUID angegebenen Kennung und zuletzt unter der Kennung $TSOS gesucht. Da das Privileg OPERATING das Privileg TSOS nicht mehr voraussetzt, muss die RUN-Datei ggf. „shareable“ gemacht werden.

Bei Verwendung des Systemparameters NBCONOPI=Y kann nicht jeder Operator die Kommandos AGOGO oder CANCEL-RUN-PROCESS für jede RUN-Sequenz eingeben, auch wenn er über die notwendigen Berechtigungsschlüssel verfügt.
Die beiden Kommandos sind nur dann zulässig, wenn mindestens eine der folgenden Bedingungen erfüllt ist:

- die Benutzerkennung des Aufrufers stimmt mit der Kennung überein, unter der das RUN-Kommando abgesetzt wurde
- die Benutzerkennung des Aufrufers ist TSOS
- das RUN-Kommando wurde von einem berechtigten Benutzerprogramm mit generiertem Berechtigungsnamen eingegeben
- der Aufrufer des RUN-Kommandos hat sich vom System abgemeldet

Beim Hochfahren des Systems wird die CMDFILE abgearbeitet und dabei so behandelt wie ein abgesetztes RUN-Kommando auf diese Datei von der Hauptbedienstation. Hier ist vorher kein SET-LOGON-PARAMETERS- und REQUEST-OPERATOR-ROLE-Kommando nötig, denn für diese CMDFILE und nur bei Startup wird die Berechtigungsprüfung anhand von Berechtigungsschlüsseln so durchgeführt, als wären der Hauptbedienstation alle Berechtigungsschlüssel zugewiesen. Alle aus der CMDFILE aufgerufenen RUN-Kommandos genießen ebenfalls dieses Privileg.

Das Kommando RUN und die mit ihm logisch verbundenen Kommandos (z.B. AGOGO und CANCEL-RUN-PROCESS) sind einer Benutzertask auch mit OPERATING-Privileg nicht erlaubt. Hier sind SDF-P-Prozeduren anwendbar, in denen auch Kommandos des Privilegs OPERATING benutzt werden können.

## 17.5 Ausübung von Systemverwaltungsfunktionen durch den Operator

Dem Operator an Konsolen werden über das Kommando SHOW-CMD-ATTRIBUTES alle zu diesem Zeitpunkt bekannten Kommandos genannt. Kommandos, die den Schlüssel $ haben, sind jedoch verboten (Eingabe wird abgewiesen).

In der Menge der $-Kommandos sind auch Kommandos enthalten, die keinerlei Systembetreuungsfunktion anbieten. Sollen einzelne dieser Kommandos an Konsolen erlaubt werden, müssen sie bereits zum Zeitpunkt der Systemeinleitung mit Hilfe des Parameterservice (Parametersatz OPR, siehe Seite 104) einem anderen Berechtigungsschlüssel zugeordnet werden.

Dazu ist für jedes Kommando eine SET-CMD-CODE-Anweisung erforderlich.
Der Berechtigungschlüssel ist von der Systembetreuung festzulegen.

*Beispiel*

Das Kommando mit der Bezeichnung SHOW-FILE-ATTRIBUTES soll dem Aufgabengebiet „allgemeine Aufgaben“ (Berechtigungsschlüssel E) zugeordnet werden.
Die Bedienplätze K1 und K2 sollen „allgemeine Aufgaben und Befugnisse“ ausführen dürfen.

```
...
SET-CMD-CODE AUTHORIZATION-CODE=E,CMD-NAME=SHOW-FILE-ATTRIBUTES
SET-CODE CODE=E,CONSOLE=(K1,K2)
...
```

Folgende Kommandos werden vom Hersteller standardmäßig mit dem Berechtigungsschlüssel $ ausgeliefert:

```
ADD-FILE-LINK
ADD-MASTER-CATALOG-ENTRY
ADD-PASSWORD
ADD-USER

COPY-FILE
CREATE-FILE
CREATE-FILE-GENERATION
CREATE-FILE-GROUP
CREATE-JV
CREATE-TAPE-SET

DELETE-FILE
DELETE-FILE-GENERATION
DELETE-FILE-GROUP
DELETE-JV
DELETE-SYSTEM-FILE
DELETE-TAPE-SET

EXPORT-FILE
EXTEND-TAPE-SET

IMPORT-FILE

LOCK-USER

MODIFY-FILE-ATTRIBUTES
MODIFY-FILE-GENERATION-SUPPORT
MODIFY-FILE-GROUP-ATTRIBUTES
MODIFY-JV-ATTRIBUTES
MODIFY-MASTER-CATALOG-ENTRY
MODIFY-SPACE-SATURATION-LEVELS
MODIFY-USER-ATTRIBUTES
MODIFY-USER-PUBSET-ATTRIBUTES

PRINT-DOCUMENT

REMOVE-JV-LINK
REMOVE-MASTER-CATALOG-ENTRY
REMOVE-PASSWORD
REMOVE-USER

SET-JV-LINK
SHOW-FILE-TRANSFER
SHOW-JV-ATTRIBUTES
SHOW-JV-LINK
SHOW-SPOOL-CHARACTER-SETS
SHOW-SPOOL-DEVICES
SHOW-SPOOL-FORMS
SHOW-SPOOL-PARAMETERS
SHOW-USER-ATTRIBUTES

UNLOCK-USER

WRITE-SPOOL-TAPE
```

Dagegen sind die Kommandos `SHOW-FILE-ATTRIBUTES` und `SHOW-SDF-PARAMETERS` (voreingestellter Berechtigungsschlüssel @) sowie `MODIFY-SDF-PARAMETERS` (voreingestellter Berechtigungsschlüssel *) für den Operator standardmäßig verfügbar. Sie werden benötigt, um im Notfall von der Konsole aus die Syntaxdatei-Konfiguration wieder so weit herzustellen, dass neue Anmeldungen im Dialog möglich sind.

Wenn dies nicht gewünscht wird, dann können sie durch Zuweisen des Berechtigungsschlüssels $ verboten werden.

# 18 Systemzeit-Verwaltung

Auf den BS2000/OSD-Servern gibt es unterschiedliche Uhren und damit Zeiten:

- die BS2000-Systemzeit, basierend auf dem TODR (Time of Day Register);
  Auf S-Servern ist das TODR eine eigenständige Uhr,
  auf SQ-Servern wird das TODR durch X2000 emuliert.
- die Zeit des Serviceprozessors (SVP-Zeit), die von BS2000/OSD durch den Befehl „Store Real Clock“ ermittelt werden kann;
  Auf S-Servern ist die SVP-Uhr eine eigenständige Uhr,
  auf SQ-Servern wird sie von X2000 emuliert und mit der Zeit des Trägersystems versorgt.
- die Zeit des Service- und Konsolprozesors (SKP-Zeit);
  nur an S-Servern;
  I.d.R. synchronisiert der SKP die SVP-Zeit mit seiner eigenen Zeit. Dies wird an der „Store-Real-Clock“-Schnittstelle angezeigt.
- die Zeit des Trägersystems;
  nur an SQ-Servern;
  Sie wird von X2000 unmittelbar als SVP-Zeit verwendet.

Die SKP-Zeit und damit auch die SVP-Zeit an S-Servern können mit externen Zeitgebern synchronisiert sein. Bei SQ-Servern gilt das Gleiche für die Zeit des Trägersystems und damit automatisch für die SVP-Zeit.

Darüber hinaus kann BS2000/OSD in einen Rechnerverbund integriert sein, in dem die Zeit der Mitglieder synchronisiert wird.

Das vorliegende Kapitel erläutert diese Zusammenhänge. Es gibt einen Überblick über die Verwaltung der Systemzeit in BS2000/OSD und beinhaltet die Initialisierung und die Synchronisation der Systemzeit sowie die Sommer-/Winterzeit-Umstellung und Spezialfälle.

Zur Zeitbestimmung bei der Systemeinleitung siehe Seite 33.

# 18.1 Systemzeit

Als Systemzeit für eine BS2000-Session dient die lokale gesetzliche Zeit. Alle Zeitstempel im privilegierten Zustand (TPR/SIH/MER) basieren auf dieser Zeit. Sie wird über die Zeit-Informationsdienste (GTIME, GDATE, GETOD) an Benutzerprogramme (TU) geliefert.

Einige Datumsangaben (insbesondere die Zeitstempel in den Dateikatalogen) erfolgen in UTC-Zeit. Einige HW-nahe, performance-kritische Komponenten nutzen den mit STCK (Store Clock) ermittelten TODR-Wert.

Die aktuelle Systemzeit wird folgendermaßen berechnet:

- Ermittlung des aktuellen STCK-Wertes
- Subtraktion des TODR-Korrekturwertes, resultierend aus Zeitkorrekturen von abgeschlossenen Synchronisationsaufträgen und dem initialen IPL-Korrekturwert
- Berücksichtigung eines aktuell aktiven Synchronisationsauftrages

Ein durch STCK ermittelter Zeitwert stimmt also nicht mit demjenigen der offiziellen Informationsdienste überein und gibt somit nicht die aktuelle Zeit wieder.

Mit dem Kommando SHOW-SYSTEM-INFORMATION INFORMATION=*SYSTEM-TIME-PARAMETER können Informationen über die Parameter der Zeiteinstellung eingeholt werden.

## 18.1.1 TODR als HW-Uhr

Das Time-of-Day-Register (TODR) ist ein 64 Bit breites Register mit folgenden Eigenschaften:

- Es wird im Bereich [Bit 63, Bit 12] pro Mikrosekunde um den Wert 1 erhöht wird und realisiert damit eine Uhr.
- Es kann durch den privilegierten Befehl Set Clock (SCK) geändert werden.
- Es kann durch den nichtprivilegierten Befehl Store Clock (STCK) gelesen werden.

Durch die Breite der Uhr im TODR ist auch die Größe der darstellbaren Zeitspanne begrenzt. BS2000/OSD ab V9.0 erweitert die Zeitdarstellung im TODR kompatibel durch so genannte Epochen, siehe Abschnitt „TODR-Epochen“ auf Seite 707.

Während IPL wird das TODR initialisiert. In der Standard-Epoche ist dies gleichbedeutend mit dem Wert „Anzahl Mikrosekunden seit 1.1.1900 00:00:00“. Für andere Epochen muss das TODR gemäß der Epoche interpretiert werden, um einen korrekten Zeitwert zu erhalten, siehe Abschnitt „TODR-Epochen“ auf Seite 707.

### Virtuelles TODR auf S-Servern unter VM2000

Die VM2000-Firmware erlaubt, für jede VM ein eigenes virtuelles TODR zu führen.
Das HW-TODR der S-Server steht nur dem Hypervisor zur Verfügung. Alle TODRs takten gleich. Zum HW-Kontext einer VM gehört ein VM-spezifischer Korrekturwert des virtuellen TODR gegenüber dem HW-TODR.

Der Hypervisor initialisiert bei S-Servern das HW-TODR mit dem Wert „Monitorzeit – 24 Stunden“.

### Emuliertes TODR auf SQ-Servern

Auf SQ-Servern werden Set Clock und Store Clock von der Firmware emuliert. Das emulierte TODR taktet unabhängig von der Zeit des Trägersystems.
Alle VM2000-Gastsysteme haben ein eigenes virtuelles TODR.

## 18.1.2 TODR-Epochen

BS2000/OSD bis V8.0 kennt nur eine Form des TODR. Damit kann die Systemzeit vom 1.1.1900 00:00:00 Uhr bis 17.9.2042 23:53:47.370495 Uhr dargestellt werden.

BS2000 ab V9.0 erweitert das TODR um so genannte TODR-Epochen. Damit kann eine Systemzeit bis 18.3.4317 02:44:48.587775 Uhr dargestellt werden.

Die TODR-Epoche für den aktuellen Systemlauf wird im Startup-Parameterservice, Parametersatz GTIME, Parameter EPOCH eingestellt, siehe Abschnitt „Steuerung der Systemzeit (GTIME)“ auf Seite 87. Sie kann während des Systemlaufs nicht verändert werden.

Der Parameter EPOCH (auch als „Epoch Designator“ bezeichnet) wird in den Benutzer-Makros CTIME (Rechnen mit Zeitstempeln) und GTIME (Datum und Uhrzeit anfordern) automatisch berücksichtigt, siehe Handbuch „Makroaufrufe an den Ablaufteil [30]. GTIME (und auch das Kommando SHOW-SYSTEM-INFORMATION) liefern den aktuellen EPOCH-Wert als Rückkehrinformation. Programme, die selbst mit dem TODR rechnen, können den im Abschnitt „Rechnen mit TODR-Epochen“ auf Seite 711 beschriebenen Algorithmus verwenden.

Der Parameter EPOCH besteht aus zwei Sedezimalziffern <epc><epo>:

- die erste Sedezimalziffer (<epc>, „Epoch Counter“) bezeichnet aufsteigend eine der disjunkten Haupt-Epochen des TODR
- die zweite Sedezimalziffer (<epo>, „Epoch Offset“) beschreibt aufsteigend eine TODR-Epoche, die in der durch <epc> bestimmten Haupt-Epoche beginnt und in die nachfolgende Haupt-Epoche hineinreicht, also die Haupt-Epochen überlappt
- EPOCH=00 bezeichnet die Standard-Epoche, d.h. den Zeitraum von 1.1.1900 00:00:00 Uhr bis 17.9.2042 23:53:47.370495 Uhr

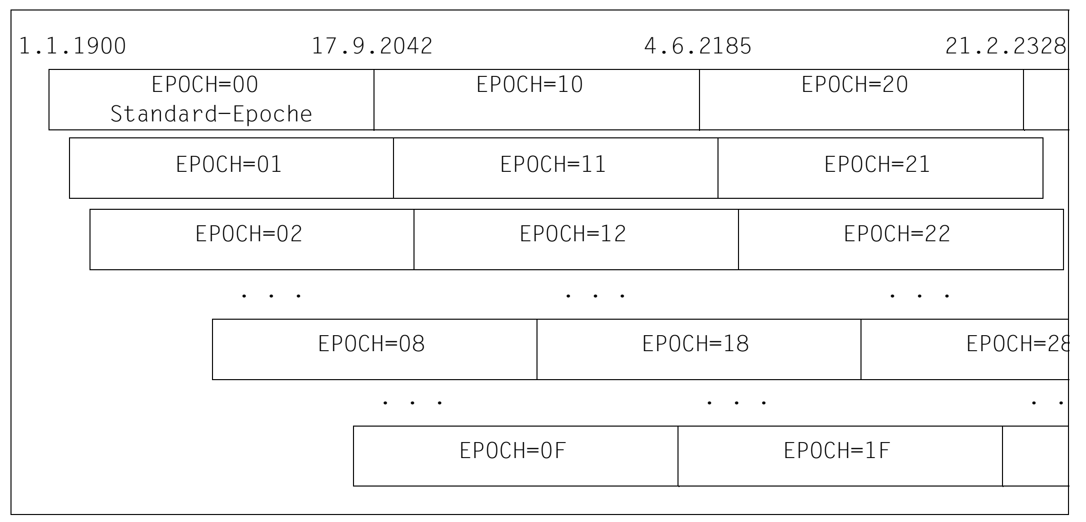



Bild 38: TODR-Epochen

Auf diese Weise ist für ein System ein gleitender Übergang zwischen den Epochen möglich, siehe das folgende Beispiel.

### Beispiel

Bisher gelte die Standard-Epoche (bis 17.9.2042 23:53:47.370495 Uhr), also EPOCH=00.

Nun sei das aktuelle Tagesdatum der 2.1.2030 und die Systembetreuung sehe die Notwendigkeit, das System auf den Ablauf der Standard-Epoche am 17.9.2042 vorzubereiten.

Dies könnte z.B. mit der Einstellung EPOCH=08 geschehen. Damit wird für das TODR der Zeitraum vom 11.5.1971 11:56:53.685248 Uhr bis 26.1.2114 11:50:41.055743 eingestellt, siehe nachfolgende Tabelle. Zeitstempel vor dem 11.5.1971 11:56:53.685248 Uhr können dann nicht mehr ohne Umrechnung mit dem aktuellen TODR verglichen werden.

Theoretisch könnten bei Umstellung am 2.1.2030 die Werte EPOCH=01 bis EPOCH=0E verwendet werden. Niedrige EPOCH-Werte legen aber eine Epoche fest, die die bisherige Epoche um nur wenige Jahre überschreitet. Bei hohen Epoch-Werten besteht die Gefahr, dass Zeitstempel, die nur wenige Jahre zurückliegen, nicht mehr direkt mit dem aktuellen TODR verglichen werden können.

### Tabellen zu den TODR-Epochen

| EPOCH= | Datum und Uhrzeit | | Wert des TODR | |
|---|---|---|---|---|
| | von | bis | von | bis |
| **00** | 1.1.1900<br>00:00:00.000000 | 17.9.2042<br>23:53:47.370495 | 00000000 00000000 | FFFFFFFF FFFFF000 |
| **01** | 2.12.1908<br>19:29:36.710656 | 19.8.2051<br>19:23:24.081151 | 10000000 00000000 | 0FFFFFFF FFFFF000 |
| **02** | 3.11.1917<br>14:59:13.421312 | 20.7.2060<br>14:53:00.791807 | 20000000 00000000 | 1FFFFFFF FFFFF000 |
| **03** | 5.10.1926<br>10:28:50.131968 | 21.6.2069<br>10:22:37.502463 | 30000000 00000000 | 2FFFFFFF FFFFF000 |
| **04** | 6.9.1935<br>05:58:26.842624 | 23.5.2078<br>05:52:14.213119 | 40000000 00000000 | 3FFFFFFF FFFFF000 |
| **05** | 7.8.1944<br>01:28:03.553280 | 24.4.2087<br>01:21:50.923775 | 50000000 00000000 | 4FFFFFFF FFFFF000 |
| **06** | 8.7.1953<br>20:57:40.263936 | 24.3.2096<br>20:51:27.634431 | 60000000 00000000 | 5FFFFFFF FFFFF000 |
| **07** | 9.6.1962<br>16:27:16.974592 | 24.2.2105<br>16:21:04.345087 | 70000000 00000000 | 6FFFFFFF FFFFF000 |
| **08** | 11.5.1971<br>11:56:53.685248 | 26.1.2114<br>11:50:41.055743 | 80000000 00000000 | 7FFFFFFF FFFFF000 |
| **09** | 11.4.1980<br>07:26:30.395904 | 28.12.2122<br>07:20:17.766399 | 90000000 00000000 | 8FFFFFFF FFFFF000 |
| **0A** | 13.3.1989<br>02:56:07.106560 | 29.11.2131<br>02:49:54.477055 | A0000000 00000000 | 9FFFFFFF FFFFF000 |
| **0B** | 11.2.1998<br>22.25:43.817216 | 29.10.2140<br>22:19:31.187711 | B0000000 00000000 | AFFFFFFF FFFFF000 |
| **0C** | 13.1.2007<br>17:55:20.527872 | 30.9.2149<br>17:49:07.898367 | C0000000 00000000 | BFFFFFFF FFFFF000 |
| **0D** | 15.12.2015<br>13:24:57.238528 | 1.9.2158<br>13:18:44.609023 | D0000000 00000000 | CFFFFFFF FFFFF000 |
| **0E** | 15.11.2024<br>08:54:33.949184 | 3.8.2167<br>08:48:21.319679 | E0000000 00000000 | DFFFFFFF FFFFF000 |
| **0F** | 17.10.2033<br>04:24:10.659840 | 4.7.2176<br>04:17:58.030335 | F0000000 00000000 | EFFFFFFF FFFFF000 |

Tabelle 53: alle TODR-Epochen mit <epc> = 0

| EPOCH= | Datum und Uhrzeit | | Wert des TODR | |
|---|---|---|---|---|
| | von | bis | von | bis |
| **00** | 1.1.1900<br>00:00:00.000000 | 17.9.2042<br>23:53:47.370495 | 000000 00000000 | FFFFFF FFFFF000 |
| **10** | 17.9.2042<br>23:53:47.370496 | 4.6.2185<br>23:47:34.740991 | 000000 00000000 | FFFFFF FFFFF000 |
| **20** | 4.6.2185<br>23:47:34.740992 | 21.2.2328<br>23:41:22.111487 | 000000 00000000 | FFFFFF FFFFF000 |
| **...** | | | | |
| **F0** | 12.9.4040<br>22:26:50.557440 | 31.5.4183<br>22:20:37.927935 | 000000 00000000 | FFFFFF FFFFF000 |

Tabelle 54: Haupt-Epochen des TODR (<epc> = 0, 1, 2 ...)

### TODR-Format TODX

In BS2000/OSD ab V9.0 gibt es ein neues Format TODX für das TODR. TODX besteht aus einem Doppelwort und enthält die Anzahl der Mikrosekunden seit Beginn der Standard-Epoche (1.1.1900 00:00:00 Uhr).

Das Format TODX wird in den Benutzer-Makros CTIME (Rechnen mit Zeitstempeln) und GTIME (Datum und Uhrzeit anfordern) verwendet bzw. geliefert, siehe Handbuch „Makroaufrufe an den Ablaufteil [30]. Es dient auch zur Epochen-unabhängigen Darstellung eines Zeitstempels und zum Rechnen mit Zeitstempeln aus unterschiedlichen Epochen.

Innerhalb der Standard-Epoche entspricht TODX einem TODR, dessen Inhalt um 12 Bit nach rechts geschoben ist. Im Gegensatz zum TODR (Obergrenze 17.9.2042 23:53:47.370495 Uhr) reicht die obere Grenze von TODX jedoch weit in die Zukunft (Obergrenze 18.3.4317 02:44:48.587775). Diese Obergrenze beruht auf dem inneren Zusammenhang mit dem um EPD erweiterten TODR-Format.

### Rechnen mit TODR-Epochen

Folgender Algorithmus berechnet aus einem gegebenen TODR und dem zugehörigen Wert des EPD die Anzahl der Mikrosekunden seit Beginn der Standard-Epoche (1.1.1900, 0 Uhr), also das TODX-Format.

```
:* todx: Anzahl der Mikrosekunden seit 1.1.1900 00:00:00.000000 Uhr
:* todr: Wert des TODR, geliefert von GTIME oder CTIME
:* epd : Wert des EPD, geliefert von GTIME
:* epc : Epoch-Counter (erste Sedezimalziffer des EPD)
:* epo : Epoch-Offset (zweite Sedezimalziffer des EPD)

epc:=epd >> 4;
:* epc wird gebildet aus epd, um 4 Bit nach rechts verschoben

epo:=epd & x'0F';
:* epo wird gebildet aus epd, linke 4 Bit gelöscht

todx:=todr >> (3*4);
:* todx wird initialisiert aus gegebenem todr, um 12 Bit nach rechts
:* verschoben

if((todx >> 12*4) LT epo) epc:=epc+1;
:* wenn das initialisierte todx (verschoben um weitere 48 Bit nach rechts)
:* kleiner ist als epo, dann liegt die Zeit in der nächsten Haupt-Epoche;
:* epc muss also um eins erhöht werden.

todx:=todx + epc * x'00100000 00000000';
:* korrektur des initialisierten todx um den Wert von epc
```

Der höchste mögliche Wert für TODX ergibt sich für `epd=x'FF'` und `todr=x'EFFFFFFF FFFFF000'` zu `todx=x'010EFFFF FFFFFFFF'`.

Weiterhin gilt für eine gegebene Epoche EPD:

– Epochenbeginn: `TODX = EPC << (13*4) + EPO << (12*4)` (einschließlich)

– Epochenende: `TODX = (EPC+1) << (13*4) + EPO << (12*4)` (ausschließlich)

*Anmerkung*

Ein Zeitstempel `todr` der Standard-Epoche, wie er möglicherweise in „alten" Datenbeständen noch existiert, kann mit obigem Algorithmus einfach in das TODX-Format gebracht werden und dann mit einem TODX-Wert einer beliebigen Epoche verglichen werden. Da in diesem Fall `epd=0` ist, reduziert sich der Algorithmus auf: `todx:=todr >> (3*4)`.

### 18.1.3 TODR-Korrekturwerte

TODR-Korrekturwerte des Zeitmanagements haben folgende Ziele:

1. Die Systemzeit kann ohne Änderung des TODR an externe Zeitgeber bzw. an eine Verbundzeit angepasst werden; damit ist ein Rückstellen der Systemzeit möglich.
2. Bei Sommer-/Winterzeitumstellung ermöglicht dies einen Sprung der Systemzeit ohne Beeinflussung des TODR.

Insbesondere ein Rückstellen des TODR verbietet sich im laufenden System, da viele Algorithmen im Betriebssystem wie in der Benutzersoftware die Monotonie des TODR unterstellen.

Nach jedem Abschluss eines Synchronisationsauftrages der Systemzeit wird der aktuell gültige TODR-Korrekturwert um diesen Auftragswert bei einem Rückstellen erhöht bzw. bei einem Vorstellen erniedrigt. Angemeldete Realzeit-Timer (STXIT bzw. systemintern) werden entsprechend angepasst. Bei der Bereitstellung der aktuellen Systemzeit im Rahmen von GTIME, GDATE oder GETOD wird der aktuelle TODR-Korrekturwert berücksichtigt.

### 18.1.4 Synchronisation der Systemzeit mit externen Zeitgebern oder im Verbund

BS2000/OSD bietet eine privilegierte sowie eine POSIX-Schnittstelle „ADJUST-TIME“ zur Anpassung der Systemzeit (nicht des HW-TODR!) an externe Uhren. Diese Schnittstelle wird genutzt für:

- Synchronisation in einem NTP-Verbund
- Synchronisation mit einem XCS-Verbund
- Synchronisation mit der SVP/SKP-Zeit (bei S-Servern) bzw. mit der SVP/Trägersystem-Zeit (bei SQ-Servern)

Ein solcher Synchronisationsauftrag führt für einen Zeitraum, der viermal so lang ist wie der geforderte Anpassungswert, zu einem langsameren bzw. schnelleren Takten der Systemzeit. Hierbei wird die Nebenbedingung eingehalten, dass zwei aufeinander folgende GTIME-Aufrufe einer Task weiterhin monoton steigende Zeitwerte liefern.

Während der Anpassungsphase wird die zu berechnende Systemzeit um den bereits durchgeführten Synchronisationsanteil zusätzlich korrigiert:

$$\frac{t - t_a}{t_e - t_a} \times \Delta$$

Dabei bedeutet:

| | |
|---|---|
| $t$ | aktuelle Zeit |
| $t_a$ | Zeitpunkt des Synchronisationsbeginns |
| $t_e$ | Zeitpunkt des Abschlusses der Synchronisation |
| $\Delta$ | Anpassungswert |

Erst mit Abschluss der Synchronisation wird der TODR-Korrekturwert um den gewünschten Anpassungswert korrigiert.

Zwischen den einzelnen Instanzen, die Synchronisationsaufträge stellen dürfen, ist eine Prioritätenreihenfolge entsprechend folgender Auflistung festgelegt. Sind mehrere Instanzen aktiv, werden nur Aufträge derjenigen mit höchster Priorität ausgeführt.

Die Prioritäten sind in folgender absteigenden Reihenfolge festgelegt
(1. = höchste Priorität, 2. = zweithöchste Priorität, usw.):

1. BS2000/OSD ist Teilnehmer eines NTP-Verbundes, der mit einem externen Zeitgeber konfiguriert ist.
2. Server mit Anschluss an externe Zeitgeber
   - S-Server: Eine vom SKP 3970 synchronisierte SVP-Uhr, wobei der SKP 3970 einen Anschluss an einen externen Zeitgeber besitzt.
   - SQ-Server: Die Zeit des Trägersystems ist mit einem externen Zeitgeber synchronisiert.
3. BS2000/OSD ist Teilnehmer eines XCS-Verbundes.
4. BS2000/OSD ist Teilnehmer eines NTP-Verbundes, der ohne externen Zeitgeber konfiguriert ist.
5. Server ohne Anschluss an externe Zeitgeber
   - S-Server: Eine vom SKP 3970 synchronisierte SVP-Uhr, wobei der SKP 3970 keinen Anschluss an einen externen Zeitgeber besitzt.
   - SQ-Server: Eine mit der Zeit des Trägersystems synchronisierte SVP-Uhr, wobei die Zeit des Trägersystems nicht mit einem externen Zeitgeber synchronisiert ist.

i Bei der Zeitsynchronisation auf Basis der SVP-Uhr (Punkte 2 und 5) ist zu beachten, dass die SVP-Uhr nur eine Auflösung im Sekundenbereich hat. Dadurch kann es zwischen SVP-Uhr und BS2000-Zeit sowie an S-Servern zusätzlich auch zwischen SVP-Uhr und SKP-Zeit jeweils zu Abweichungen von ± 1 Sekunde kommen. In Summe ergibt sich damit für die BS2000-Systemzeit an S-Servern eine maximale Abweichung von ± 2 Sekunden zur Zeitquelle, an SQ-Servern von ± 1 Sekunde.

In einem NTP- oder XCS-Verbund (Punkte 1, 3 und 4) erfolgt die Übertragung und Synchronisation der Zeit mit wesentlich höherer Auflösung. Die Abweichung der BS2000-Systemzeit von der maßgebenden Zeitquelle liegt im Millisekunden-Bereich.

### 18.1.5 Subsystem GET-TIME

BS2000/OSD bietet mehrere Schnittstellen zur Ermittlung der aktuellen Systemzeit: GTIME, GDATE, GETOD. Laufzeitroutinen von Hochsprachen mit analoger Semantik setzen entsprechend darauf auf.

GTIME bietet die Systemzeit (lokale Zeit sowie UTC-Zeit) im Binär-, ISO4- und TODR/TODX-Format an. Es ist zu beachten, dass sich die Systemzeit im TODR-Format wegen der TODR-Korrekturwerte von einem gleichzeitig ermittelten STCK-Wert unterscheidet.

Der Server für die langfristig garantierte Schnittstelle GTIME ist sowohl als Bestandteil von BS2000/OSD-BC als auch als nichtprivilegiertes TU-Subsystem GET-TIME realisiert, um bei Aufrufen aus TU eine SVC-freie, performante Bearbeitung zu bieten. Der GTIME-Aufruf aus TU hat für die Bereitstellung der Systemzeit im TODR-Format eine Pfadlänge von weniger als 200 Befehlen. Der Einsatz von TASKDATE (siehe Seite 730) bzw. die Ermittlung der XCS-Zeit (siehe Seite 723) führt zu einer Bearbeitung im TPR. Das Subsystem enthält weiterhin den Server für CTIME, der Makroschnittstelle zur Arithmetik mit Zeitstempeln und zur Konvertierung von Zeitstempeln bezüglich Format und Zeitbasis. Die Datenbasis für die beiden Dienste (TODR-Korrekturwerte, Daten eines aktiven Synchronisationsvorganges sowie Sommer-/ Winterzeit-Umstellzeitpunkte) sind sowohl für den Betriebssystem-Modul im Systemadressraum als auch TU-schreibgeschützt im Subsystem abgelegt.

Das Subsystem GET-TIME ist obligatorisch, wird vor „System Ready“ automatisch gestartet und besitzt zwei, über die explizite Auswahl eines der zwei ausgelieferten Subsystemkataloge gesteuerte Ausprägungen:

- Laden unterhalb 16 MB (Standard), falls in dem System Nicht-XS-fähige Programme im Einsatz sind
- Laden oberhalb 16 MB (zur Entlastung des unteren Adressraumes), falls nur XS-fähige Programme das Subsystem nutzen

Neben dem Subsystem GET-TIME gibt es auf SQ-Servern noch das nur Betriebssystem-intern genutzte Subsystem GET-TIMX.

### GET-TIME Startup-Parameter

Die Startup-Parameterdatei enthält einen Abschnitt für GET-TIME (Parametersatz GTIME). Dieser beschreibt für eine oder mehrere Zeitzonen folgende Information:

- Information über die Zeitzone (d.h. Differenz zur UTC-Zeit)
- Sommerzeit-Information (Differenz Sommerzeit / Winterzeit, vergangene und zukünftige Umstellzeitpunkte)

Diese Information wird benötigt für

- Zeitkonvertierungen Lokale-Zeit / UTC (und umgekehrt) für beliebige Zeitstempel durch die Funktion CTIME. Je mehr Umstellzeitpunkte verfügbar sind, um so größer ist der Zeitbereich, in dem CTIME eine korrekte Umrechnung Lokale-Zeit / UTC vornehmen kann.
- Bestimmung von Sommerzeit / Winterzeit bei Startup
- Umstellung der Systemzeit zum nächsten Termin einer Sommerzeit / Winterzeit Umstellung

Bei S-Servern ist die Zeitzone des Standortes des Servers über SVP-Frame einzustellen. Bei SQ-Servern ist diese Information im Trägersystem bekannt. Startup ermittelt sich dieses Datum und wählt damit vorrangig den korrespondierenden Zeitzonen-Parameterabschnitt aus.

Für weitere Hinweise zum Startup-Parameter GTIME, insbesondere zum Umfang der bereitgestellten Umstellungszeitpunkte, siehe Abschnitt „Steuerung der Systemzeit (GTIME)“ auf Seite 87.

## 18.2 SVP-Zeit

Der SVP hat aus BS2000-Sicht eine eigene Uhr, die über die privilegierte Schnittstelle „Store Real Clock“ vom Betriebssystem abfragbar ist. Die SVP-Uhr muss immer in lokaler Zeit laufen. „Store Real Clock“ stellt die SVP-Zeit mit Sekunden-Auflösung bereit. Die „Store Real Clock“-Zusatzinformation ist abhängig vom Server-Typ:

| Server-Typ | Zusatzinformation |
|---|---|
| S-Server | Zeitzone,<br>Anzeige:<br>– Synchronisation durch den SKP 3970 aktiv / ausgefallen<br>– SKP 3970 mit externem Zeitgeber synchronisiert |
| SQ-Server | Zeitzone,<br>Anzeige: mit externem Zeitgeber synchronisiert |

Modellabhängig ist diese Uhr als Hardware vorhanden oder durch X2000 emuliert, sie taktet jedoch in beiden Fällen unabhängig vom TODR.

| Server-Typ | Art der Einbettung der SVP-Uhr |
|---|---|
| S-Server | eigenständige SVP-Uhr vorhanden; SKP 3970 synchronisiert diese mit der SKP-Uhr; die SKP-Uhr kann mit einer Funkuhr oder mit anderen Systemen mittels NTP synchronisiert sein. |
| SQ-Server | SVP-Uhr wird von X2000 mit Hilfe der Zeit des Trägersystems emuliert; die Zeit des Trägersystems kann mittels NTP mit einem externen Zeitgeber synchronisiert sein. |

### Hinweise zum Einsatz unter VM2000

Auf S-Servern wird unter VM2000 der „Store Real Clock“ eines Gastsystems vom Hypervisor emuliert. Im Normalfall wird hierzu die Systemzeit des Monitorsystems herangezogen. Es gibt folgende Ausnahmen, die zur Ausführung eines „Store Real Clock“-Befehls führen:

- Das Monitorsystem wurde mit einer expliziten vom Operator eingegebenen Zeit gestartet.
- Das Monitorsystem nimmt an einem XCS-Verbund teil.
- Das Monitorsystem nimmt an einem NTP-Verbund teil, der nicht mit einem externen Zeitgeber konfiguriert ist.

Auf SQ-Servern wird der „Store Real Clock“ auch unter VM2000 wie native von X2000 emuliert.

## 18.3 Initialisierung der Systemzeit

Die Systemzeit wird während IPL initialisiert, indem das TODR geladen wird.

**Zeitquellen für Systemzeit-Initialisierung**

Als Zeitquellen für die Initialisierung wird die über die Schnittstelle „Store Real Clock“ ermittelte SVP-Zeit übernommen.

Weiter gibt es folgenden Sonderfall:
Das System wird im DIALOG-Modus mit Option UNLOCK hochgefahren und der Operator gibt eine explizite Zeitangabe ein.

Die verwendete Zeitquelle wird als Insert der Startup-Meldung `NSI1163` protokolliert.

Die Initialisierung der Sommerzeit-/Winterzeit-Einstellung erfolgt mittels der SVP-Information. Liegt dort die Information nicht vor oder wurde eine manuelle Zeitangabe (mittels UNLOCK-Option) gemacht, kann eine doppelte Stunde bei Sommer-/Winterzeitwechsel nicht eindeutig ermittelt werden. Startup unterstellt dann Winterzeit.

Zur Initialisierung der Systemzeit siehe auch Abschnitt „Zeitbestimmung bei der Systemeinleitung“ auf Seite 33.

# 18.4 Synchronisation der Systemzeit

## 18.4.1 Synchronisation mit der SVP/SKP-Uhr bei S-Servern

Der SKP 3970 synchronisiert normalerweise die SVP-Uhr auf seine eigene Zeit (siehe Tabellen auf Seite 717).
Falls keine andere Synchronisations-Instanz in BS2000/OSD vorhanden ist, findet somit eine Synchronisation der Systemzeit mit der SVP-Uhr und damit implizit mit der SKP-Zeit statt. Dazu wird periodisch in BS2000/OSD ein Zeitvergleich SVP-Zeit gegen BS2000-Systemzeit vorgenommen und eine entsprechende Anpassung der BS2000-Systemzeit eingeleitet.

► In der Ausgabe des Kommandos SHOW-SYSTEM-INFORMATION findet sich im Abschnitt SYSTEM-TIME-PARAMETER die entsprechende Information:
`SYNCHRONIZATION=SKP-X`.

Für S-Server kann der Anschluss eines NTP-Servers über LAN am SKP 3970 erfolgen. Die Modelle SKP 3970-2x und -3x unterstellen grundsätzlich, dass der NTP-Server mit der gesetzlichen Zeit synchronisiert ist, und zeigen dem BS2000-System die Existenz eines externen Zeitgebers an. Ab SKP 3970-4x gilt dies nur noch, wenn die NTP-Zeit ein *stratum* ≤ 4 hat.

► In der Ausgabe des Kommandos SHOW-SYSTEM-INFORMATION findet sich im Abschnitt SYSTEM-TIME-PARAMETER die entsprechende Information:
`SYNCHRONIZATION=SERVER-CONN-EXT-REF`.

Zusätzlich kann ein ein NTP-Verbund mit externem Zeitgeber konfiguriert werden, der dann die höchstwertige Synchronisationsinstanz darstellt (siehe Abschnitt „Synchronisation in einem NTP-Verbund“ auf Seite 721).

## 18.4.2 Synchronisation mit der Zeit des Trägersystems bei SQ-Servern

Wenn keine andere Synchronisations-Instanz in BS2000/OSD vorhanden ist, dann findet auf SQ-Servern automatisch eine Synchronisation der Systemzeit mit der SVP-Uhr, also mit der Zeit des Trägersystems Linux statt. Ist die Zeit des Trägersystems nicht mit einem externen Zeitgeber synchronisiert, können alle sonstigen Synchronisationsinstanzen in BS2000/OSD (NTP oder XCS) zur Wirkung kommen.

► In der Ausgabe des Kommandos SHOW-SYSTEM-INFORMATION findet sich im Abschnitt SYSTEM-TIME-PARAMETER die entsprechende Information: `SYNCHRONIZATION=X2000`.

Ist zusätzlich der Anschluss eines externen Zeitgebers gewünscht, so ist dieser über den SQ-Manager des Servers für die Linux-Zeit zu konfigurieren.

Diese externen Zeitgeber werden in der BS2000-Zeit nur dann sichtbar, wenn ihre Qualität bis zum Trägersystem Linux erhalten bleibt (d.h.: *stratum* des NTP-Servers im Trägersystem Linux ≤ 4) .

► In der Ausgabe des Kommandos SHOW-SYSTEM-INFORMATION findet sich im Abschnitt SYSTEM-TIME-PARAMETER die entsprechende Information: `SYNCHRONIZATION=SERVER-CONN-EXT-REF`.

### 18.4.3 Synchronisation in einem NTP-Verbund

Das Software-System NTP (Network Time Protocol) organisiert für die an einem derartigen Verbund beteiligten Systeme eine gemeinsame Zeit. BS2000/OSD kann über POSIX Teilnehmer eines solchen Verbundes sein.
Jeder Verbundteilnehmer fragt die Zeit ausgewählter Systeme unter Berücksichtigung von Übertragungslaufzeiten ab und ermittelt daraus eine Verbundzeit, auf welche er seine Systemzeit synchronisiert. NTP nutzt dazu die POSIX-Schnittstellenausprägung von „ADJUST-TIME“. Server mit externen Zeitgebern können als so genannte „Time-Server“ deklariert werden, die ihre Systemzeit nicht an eine Verbundzeit anpassen, sondern umgekehrt: deren Zeit dient für die anderen Verbundteilnehmer als ausschließliche Quelle zur Ermittlung der Verbundzeit.

In BS2000/OSD können folgende zwei NTP-Ausprägungen bezüglich der Priorisierung als Synchronisationinstanz der Systemzeit zum Einsatz kommen. Die Unterscheidung erfolgt anhand der NTP-Zeitqualität *stratum*.

- STD-Einsatz
  Wenn die NTP-Zeit ein *stratum* > 4 hat, dann nimmt die BS2000-Zeitverwaltung an, dass der NTP-Server nicht über eine zuverlässige Zeitquelle verfügt, und weist ihm eine entsprechend niedrige Priorität zu. Wenn eine höher-priore Zeitquelle (Zeitgeber an SKP/X2000 oder XCS-Verbund, siehe Abschnitt „Synchronisation der Systemzeit mit externen Zeitgebern oder im Verbund“ auf Seite 713) existiert, dann lässt die BS2000-Zeitverwaltung keine Synchronisationsaufträge von NTP zu. Nur wenn keine höherpriore Zeitquelle existiert, dann wird die BS2000-Systemzeit mit der NTP-Zeit synchronisiert.

  ► In der Ausgabe des Kommandos SHOW-SYSTEM-INFORMATION findet sich im Abschnitt SYSTEM-TIME-PARAMETER die entsprechende Information:
  `SYNCHRONIZATION=DCE/NTP`.

- NTP-Verbund mit Anschluss an einen externen Zeitgeber
  Wenn die NTP-Zeit ein *stratum* ≤ 4 hat, dann nimmt die BS2000-Zeitverwaltung an, dass der NTP-Server über eine zuverlässige Zeitquelle verfügt und weist ihm höchste Priorität zu. Die BS2000-Systemzeit wird mit der NTP-Zeit synchronisiert. Die Genauigkeit der NTP-Zeit liegt im Millisekunden-Bereich.

  ► In der Ausgabe des Kommandos SHOW-SYSTEM-INFORMATION findet sich im Abschnitt SYSTEM-TIME-PARAMETER die entsprechende Information:
  `SYNCHRONIZATION=BS2-CONN-EXT-REF`.

Eine Sommer/Winterzeit-Umstellung findet zu der durch NTP gesteuerten Systemzeit statt. NTP kann über eine solche Zeitumstellung hinweg im Einsatz bleiben.

### Hinweise zum Einsatz unter VM2000

Ein VM2000-Gastsystem (auch das VM2000-Monitorsystem) kann unabhängig von den restlichen virtuellen Maschinen eines unter VM2000 betriebenen Servers an einem NTP-Verbund beteiligt sein. Ist das Monitorsystem Teilnehmer eines derartigen Verbundes, hängt bei S-Servern die Emulation der an die Gastsysteme gelieferten SVP-Zeit („Store-Real-Clock") von der NTP-Ausprägung ab:

- Bei NTP mit Anschluss an einen externen Zeitgeber wird den Gastsystemen die Monitorzeit bereitgestellt. Die Teilnahme der Monitor-VM an einem derartigen Verbund garantiert für alle Gastsysteme ebenfalls die Verbundzeit, allerdings mit einer Genauigkeit von ± 1 Sekunde. Wird eine gleich hohe Genauigkeit gefordert, muss das Gastsystem als expliziter Verbundteilnehmer agieren.
- Bei NTP-Standardkonfiguration im Monitorsystem erhalten die Gastsysteme die echte SVP-Zeit. Demzufolge müssen bei Bedarf alle Gastsysteme Verbundteilnehmer sein.

Für SQ-Server wird die Teilnahme der Dom0 mit Linux/X2000 am NTP-Verbund mit Anschluss an einen externen Zeitgeber empfohlen. Dies garantiert die NTP-Verbundzeit für alle Gastsysteme, auch für das Monitorsystem, allerdings mit einer Genauigkeit von ± 2 Sekunden. Wenn eine höhere Genauigkeit gefordert wird, dann muss das System als expliziter Verbundteilnehmer agieren.

### 18.4.4 Synchronisation in einem XCS-Verbund

Ein XCS-Verbund ist eine Ausprägung der in BS2000/OSD verfügbaren HIPLEX-Funktionalität. Er ermöglicht insbesondere den Ablauf verteilter Anwendungen und umfasst folgende Funktionalität:

- automatische Ausfallerkennung und Rekonfiguration
- Lock-Management für verteilte Anwendungen (DLM)
- einheitliche Zeit im Verbund

Die Synchronisierung der Systemzeit im Verbund ist durch das Subsystem XCS-TIME, das beim XCS-Start initiiert wird, realisiert und basiert auf dem Konzept der XCS-Zeit, einer zweiten Zeit neben der Systemzeit. Die XCS-Zeit wird über GTIME (allerdings mit SVC) im TODR-Format und UTC-basiert bereitgestellt.

#### XCS-Zeit

Verteilte Anwendungen benötigen oft neben einer Lock-Funktion eine verbundweit-monotone Zeit, z.B. um lokal erstellte Logging-Dateien später chronologisch mischen zu können. Die XCS-Zeit garantiert bei zwei aufeinander folgenden Abfragen auf beliebigen Servern diese Monotonie, falls die Zeitermittlung unter dem gleichen DLM-Lock erfolgt:

Server A `GET-LOCK (Lock-Name=X1,exklusiv)` für Datenblock x

`GTIME(XCS-TIME)`

`RELEASE-LOCK(Lock-Name=X1)`

Server B: `GET-LOCK (Lock-Name=X1,exklusiv)` für den gleichen Datenblock x

`GTIME(XCS-TIME)`

`RELEASE-LOCK(Lock-Name=X1)`

Die beiden GTIME-Aufrufe liefern echt monotone Zeitwerte, womit die Logging-Information für Manipulationen am Datenblock x chronologisch unterschieden werden kann.

Jedes System verwaltet auf folgende Weise eine eigene XCS-Zeit:

1. Bei exklusiver Lock-Zuteilung durch ein System A, das Verwalter eines DLM-Locks ist, an ein zweites System B wird zusätzlich zur Zuteilungsquittung das um den Wert „eine Mikrosekunde“ inkrementierte Maximum aus XCS-Zeit und Systemzeit an den Lock-Requestor geschickt.
2. System B empfängt die Lock-Zuteilung von System A

   Das Maximum der mit Lock-Zuteilung gelieferten XCS-Zeit des Systems A und der XCS-Zeit des Systems B sowie dessen aktuelle Systemzeit bildet die neue aktuelle XCS-Zeit des Systems B.

3. GTIME-Aufruf zum Erhalt der XCS-Zeit durch eine Anwendung

   Das um den Wert „eine Mikrosekunde“ inkrementierte Maximum aus der aktuellen Systemzeit und der XCS-Zeit des Systems wird zur aktuellen XCS-Zeit und dem Aufrufer bereitgestellt.

Dieses Verfahren hält die XCS-Zeit aller beteiligten Systeme etwa gleich der Systemzeit des Systems mit der am weitesten vorausgehenden Zeit.

► In der Ausgabe des Kommandos SHOW-SYSTEM-INFORMATION findet sich im Abschnitt SYSTEM-TIME-PARAMETER die entsprechende Information: `SYNCHRONIZATION=XCS`.

Um auch die Systemzeiten möglichst identisch zu halten (um z.B. eine weitestgehende korrekte Chronologie von dezentralen System Logging-Daten zu erhalten), synchronisiert in allen beteiligten Systemen das Subsystem XCS-TIME die Systemzeit auf die XCS-Zeit des Systems. Die Systemzeit und die XCS-Zeit der einzelnen Server sind damit nur annähernd gleich; eine Garantie für eine maximale Abweichung gibt es nicht.

Der Eintritt eines Servers in einen Verbund wird solange verzögert, bis dessen Systemzeit an die der bereits aktiven Partnern angepasst ist. Ist der Zeitunterschied bei Eintritt größer als 15 Minuten, wird der Eintritt abgewiesen.

### Externe Zeitgeber und XCS-Verbund

Ein XCS-Verbund kann zusätzlich mit einem externen Zeitgeber synchronisiert werden. Der Zeitgeber kann an nur einen oder einer Teilmenge der Server des Verbundes angeschlossen werden. Server mit einem derartigen Zeitgeber veranlassen dann – auf Basis des Vergleichs ihrer Systemzeit mit den gelieferten XCS-Zeiten der Server ohne externen Zeitgeber – die Server zum notwendigen Anpassen ihrer Systemzeit.

In einem System eines XCS-Verbundes unterbleibt die Synchronisation auf die Zeit eines SKP 3970 bzw. eines Trägersystems ohne Anschluss an einen externen Zeitgeber. Ein solches System kann nicht gleichzeitig in einem NTP-Verbund in Standardkonfiguration sein. Aufrufe zur Zeit-Synchronisation durch NTP werden abgewiesen.

### Hinweise zum Einsatz unter VM2000

Ein VM2000-Gastsystem (auch das Monitorsystem) kann unabhängig von den restlichen virtuellen Maschinen des Servers an einem XCS-Verbund beteiligt sein. Ist das Monitorsystem Teilnehmer eines derartigen Verbundes, basiert die Emulation der an die Gastsysteme gelieferten SVP-Zeit (Store-Real-Clock) auf der tatsächlichen SVP-Zeit und nicht auf der XCS-synchronisierten Systemzeit des Monitorsystems.

### 18.4.5 Generelle Empfehlung für den XCS-Verbund

Die an einem XCS-Verbund (mit oder ohne externe Zeitgeber) beteiligten Linux-Trägersysteme (SQ-Server) und SKP-Systeme (S-Server) sollten alle in einen NTP-Verbund aufgenommen werden.

### 18.4.6 Synchronisationsanzeigen bei /SHOW-SYSTEM-INFORMATION

Die folgende Tabelle stellt die möglichen Anzeigen des Kommandos SHOW-SYSTEM-INFORMATION im Abschnitt SYSTEM-TIME-PARAMETER für den Wert von SYNCHRONIZATION auf beiden Server-Typen nach Prioritäten aufsteigend dar.
Beim Vorhandensein mehrerer Zeitgeber bestimmt die höchste Priorität unter ihnen den Wert von SYNCHRONIZATION.

| SYNCHRONIZATION = | **S-Server** | **SQ-Server** |
|---|---|---|
| *NONE | Fehlerfall:<br>SKP zur Zeit nicht verfügbar oder Differenz zwischen System- und SVP-Zeit zu groß für eine Synchronisation! | Fehlerfall:<br>SVP-Uhr zur Zeit nicht verfügbar oder Differenz zwischen System- und SVP-Zeit zu groß für eine Synchronisation! |
| X2000 | entfällt | Standardwert<br>(ohne externe Synchronisation) |
| SKP-X | Standardwert<br>(ohne externe Synchronisation) | entfällt |
| DCE/NTP | Synchronisation durch NTP-Verbund (in BS2000/OSD) | Synchronisation durch NTP-Verbund (in BS2000/OSD) |
| XCS | Synchronisation durch XCS-Verbund | Synchronisation durch XCS-Verbund |
| SERVER-CONN-EXT-REF | Externer Zeitgeber über SKP 3970 oder (für VM2000-Gastsysteme) im Monitorsystem angeschlossen | Externer Zeitgeber über das Trägersystem oder (für VM2000-Gastsysteme) im Monitorsystem angeschlossen |
| BS2-CONN-EXT-REF | Externer Zeitgeber über NTP (BS2000/OSD) angeschlossen | Externer Zeitgeber über NTP (BS2000/OSD) angeschlossen |

## 18.5 Unterbrechungsfreie Sommer-/Winterzeitumstellung

Die Zeitumstellung kann unterbrechungsfrei erfolgen, d.h., das System wird kontinuierlich über eine Zeitumstellung hinweg betrieben.

An den beiden Terminen der Zeitumstellung wird die Systemzeit entsprechend angepasst: d.h. in unserer mitteleuropäischen Zeitzone wird die Systemzeit im Frühjahr (meist am letzten Märzwochenende) um 1 Stunde vorgestellt, im Herbst (am letzten Oktoberwochenende) um 1 Stunde zurückgestellt.

Das Verstellen erfolgt in einem Schritt und damit anders als die sonstigen Zeitsynchronisationen: Das TODR wird nicht geändert, es läuft monoton weiter. Stattdessen wird der TODR-Korrekturwert entsprechend manipuliert, um die verstellte Systemzeit zu erhalten. Diese Umstellung wird dadurch ausgelöst, dass die Systemtask TIME bei Startup (und nach jeder Umstellung) einen Timer mit dem nächsten Umstelltermin aufzieht.

Umstellungszeitpunkte werden über den GTIME-Abschnitt der Startup-Parameterdatei bekanntgegeben. Fehlen dort zukünftige Umstellzeitpunkte, wird eine entsprechende Warnung ausgegeben. Umstellungszeitpunkte können auch im laufenden BS2000-Betrieb mit den Kommandos ADD-/MODIFY-/REMOVE-/SHOW-CHANGE-DATE verwaltet werden. Änderungen von Umstellungszeitpunkten durch die Kommandos müssen aber manuell in die Parameterdatei eingetragen werden, wenn sie im nächsten Systemlauf gültig sein sollen.

Falls die Anwendungs-Software noch nicht vollständig von Zeitermittlung mit STCK auf GTIME umgestellt ist, kann die Systemzeitumstellung durch Angabe von DIFF=0:00 im Parameterservice unterbunden werden. Das System muss dann neu gestartet werden.

*Hinweise*

- Die Schnittstelle CTIME stellt zusätzliche Warnungen für Eingabe- bzw. Ausgabezeitstempel in der doppelten bzw. übersprungenen Stunde bereit. Die mit GTIME gelieferte Systemzeit (lokale Zeit) macht zum Umstellungszeitpunkt einen entsprechenden Sprung. Anwendungen, die mit Zeitstempeln rechnen bzw. diese vergleichen, müssen negative Zeitsprünge der lokalen Zeit verarbeiten können. Es wird entweder eine Verfahrensumstellung auf UTC-Zeit oder die zusätzliche Mitführung der Sommerzeit-/Winterzeit-Anzeige empfohlen, welche bei GTIME/CTIME ebenfalls bereitgestellt wird. Die aktuelle Zeit darf nicht durch direkten TODR-Zugriff ermittelt werden.
- Benutzen Benutzerprogramme statt GTIME noch den STCK-Befehl und eigene Umrechnungsalgorithmen für Zeitstempel, sind sie auf die lokale Zeit im TODR angewiesen. In diesem Fall muss das System bei Zeitumstellung wie bisher neu geladen werden.
- Sind alle Benutzerprogramme schon auf GTIME zur Zeitabfrage umgestellt worden oder verrechnen sie STCK-Werte nur zu Differenzen, kann die unterbrechungsfreie Zeitumstellung genutzt werden.

### Verhalten des Job-Managements bei Zeitumstellung

- Umstellung von Winter- auf Sommerzeit
  - Termin-Jobs

    Durch die Umrechnung der Startzeit in UTC und anschließender Rückkonvertierung kommt ein um eine Stunde verschobener Startzeitpunkt zustande.

    Beispiel: ENTER <datei>,START=AT(02:30) führt zum Start um 03:30 (LTI). Diese Startzeit vererbt sich auf die Nachfolgejobs, wenn der Job der erste von mehreren Repeat-Jobs ist.
  - Repeat-Jobs mit Periode DAILY/WEEKLY:

    Repeat-Job-Wiederholungen, die in diesem Intervall starten müssten, fallen aus. Stattdessen wird der nächste Nachfolger genommen, das nächste Starten erfolgt also 1 Tag oder 1 Woche später.
  - Repeat-Jobs mit Stunden-/Minute-Perioden

    Die Startzeiten der Jobwiederholungen werden einfach durch Addition der Minuten der Periode auf die letzte Startzeit berechnet (in JMS-Zeit, UTC), sie überspringen also das Intervall einfach.

    Beispiel Startzeitpunkte: ... 01:45 , 01:55 , 03:05 , 03:15 ... (LTI)
- Umstellung von Sommer- auf Winterzeit
  - Termin-Jobs

    Die interne Umrechnung der Startzeit eines Termin-Jobs in UTC unterstellt Winterzeit bei der Umrechnung; der Start erfolgt dann entsprechend in der zweite Stunde des Intervalls.
  - Repeat-Jobs mit Periode DAILY/WEEKLY:

    Repeat-Job-Wiederholungen, die in diesem Intervall starten müssten, werden ebenso behandelt, starten also im zweiten Intervall.
  - Repeat-Jobs mit Stunden-/Minute-Perioden

    Die Startzeiten der Jobwiederholungen werden einfach durch Addition der Minuten der Periode auf die letzte Startzeit berechnet (in JMS-Zeit, also UTC), sie laufen also ggf. in jedem der beiden Intervalle.

*Hinweise zum Verhalten von AVAS bei Zeitumstellung*

- Umstellung von Winter- auf Sommerzeit

  Die für den Zeitraum 2.00 – 3.00 Uhr geplanten Netze werden plötzlich entsprechend den Einstellungen für Verspätungen (NET-DELAY-SOLUTION) gestartet und laufen dann normal ab. Kurz nach 3.00 Uhr kommt es somit vermehrt zu AVAS-Aktivitäten, da alle Netze, die in der verlorenen Stunde gestartet werden sollen, sofort behandelt werden.

- Umstellung von Sommer- auf Winterzeit

  Die (seit 2.00 Uhr) bereits gestarteten Netze laufen normal weiter, da sie unabhängig von der Zeit nach dem Vorläufer-Nachfolger-Prinzip gestartet werden. Nach dem Zurückstellen der Zeit werden bis 3.00 Uhr keine Netze gestartet. Mit dem Zeitstempel in der Journaldatei kann es u.U. beim Sortieren Probleme geben, da die Startzeiten von Nachfolgejobs vor denen der Vorläufer liegen können.

## 18.6 Systemstart mit spezieller Systemzeit

Die Systemzeit kann mit einer beliebigen Zeit initialisiert werden. Der Systemstart ist dazu im DIALOG-Modus mit der IPL-Option UNLOCK durchzuführen. Über zu beantwortende Meldungen kann der Operator eine beliebige Zeit angeben. SVP-Zeit und SKP-Zeit bzw. Trägersystem-Zeit bei SQ-Servern bleiben davon unberührt.

Unter VM2000 kann eine beliebige VM mit fiktiver Systemzeit und entsprechend versetzter UTC-Zeit betrieben werden. Allerdings muss jeder Systemstart dieses Gastsystems im DIALOG-Modus und der IPL-Option UNLOCK erfolgen.

Es findet keine Synchronisation mit einem evtl. vorhandenen externen Zeitgeber statt, wenn die manuell eingestellte Zeit mehr als 15 Minuten von der SVP-Zeit abweicht. Die Systemzeit taktet mit der HW-TODR-eigenen Qualität.
Da die veränderte Systemzeit auch für Zeitstempel im Systemkatalog genutzt wird, kann z.B. bei vorgestellter Systemzeit und anschließender Wiedernutzung des Pubsets in der Jetzt-Zeit der Dateizugriff wegen noch nicht abgelaufenem EXPIRATION-DATE verwehrt sein. Anwendungsspezifische Daten mit Zeitstempeln sind nach einer derartigen zweifachen Portierung ebenfalls beliebig inkonsistent und können zu undefiniertem Programmverhalten führen (insbesondere bei Datenbanken).

Einen speziellen Fall stellt der permanente Betrieb eines Systems mit fremder Zeitzone dar. Die Umstellung auf eine andere Zeitzone im Monitorsystem bzw. native funktioniert nur, falls im SVP bzw. im Trägersystem auch die neue Zeitzone eingestellt und anschließend der Startup mit IMPL durchgeführt wird.
Ansonsten (also inbesondere bei fremder Zeitzone für ein Gastsystem) darf das System die vom SVP bei einem Startup über „Store Real Clock“ erhaltene Zeit nicht als Systemzeit verwenden. Die korrekte, zur gewünschten Zeitzone passende Systemzeit kann nur noch durch manuelle Eingabe entstehen. Dazu muss jeder Startup dieses Systems als Dialog-Startup mit der Option UNLOCK erfolgen.
Darüber hinaus ist zu beachten:

- Enthält die GTIME-Parameterdatei nur die Informationen der gewünschten Zeitzone, kann der Makro CTIME im laufenden Betrieb keine Umrechnungen für fremde Zeitzonen durchführen.
- Enthält die GTIME-Parameterdatei Angaben zu mehreren Zeitzonen, ist nach manueller Eingabe der Systemzeit nicht ersichtlich, welche davon die richtige ist.
  Die Zeitinitialisierungsroutine führt dann mit dem Operator einen Meldungsdialog, welcher die Eingabe einer Zone, nicht aber einer Sommer-/Winterzeit-Differenz bzw. von Umstellungszeitpunkten für diese Zone, ermöglicht.
  Die eingegebene Zone wird im SVL hinterlegt, ein sinnvolles Arbeiten ist mit einem derartigen System aber nicht möglich. Erst beim nächsten Startup – natürlich im Modus DIALOG UNLOCK – wird statt der SVP-Zone die im SVL hinterlegte Zone verwendet und damit die umgestellte Parameterdatei auswertbar.

# 18.7 TASKDATE: Testen in simulierter Zeit

Das Tool TASKDATE wurde zum Test von datum- oder zeitsensitiven Anwendungen entwickelt.

Es ermöglicht einer Task, solange sie sich im nichtprivilegierten Zustand (TU) befindet, in versetzter Zeit zu laufen. Dazu werden alle Zeitabfragen aus TU (GTIME, GDATE, GETOD sowie Zeitservices der Hochsprachen – nicht STCK) mit einer vorher über das Kommando SET-TASK-CLOCK festgelegten, versetzten Zeit vom System befriedigt. Die simulierte Zeit kann optional an Folgetasks (ENTER) vererbt werden. Das System, Katalogeinträge der Tasks sowie Jobstart-Zeitpunkte basieren weiterhin auf der Systemzeit. Das Tool ist als DSSM-Subsystem sowie als Erweiterung des BS2000-Zeitmanagements, JMS, SDF-P, JV und openUTM realisiert. Das Tool kann im laufenden Betrieb mittels einer mit ausgelieferten Startprozedur gestartet werden.

# 19 Anhang

Im Anhang sind Übersichten und Tabellen zu den verschiedensten Themen zu finden. Sie sind in folgender Reihenfolge zusammengestellt.

- Klasse-2-Systemparameter (kurz: Systemparameter, ab Seite 732)
- Zeichensatz bei Ein-/Ausgabe über Bedienstation (ab Seite 766)
- Übersicht zu Testprivilegien (ab Seite 768)

Spezielle Übersichten, die sich nicht mehr in diesem Anhang befinden, finden sich an anderer Stelle oder in anderen Handbüchern:

- Gerätetyp-Tabelle, Volumetyp-Tabelle, Organisation der Plattenspeicher (siehe die Handbücher „Systeminstallation“ [57] oder „Kommandos“[27])
- Übersicht über die Kommandos mit den dazugehörenden Privilegien, soweit sie Bestandteil von BS2000/OSD-BC (nicht kostenpflichtig) und im Handbuch „Kommandos“ beschrieben sind, siehe Handbuch „Kommandos“ [27]
- Übersicht über die Operatorkommandos mit ihren Berechtigungsschlüsseln, siehe Abschnitt „Aufgabengebiete und ihre Aufteilung auf Bedienstationen“ auf Seite 642

## 19.1 Systemparameter

### Dynamisch änderbare Systemparameter

Folgende Systemparameter können unter der Benutzerkennung TSOS mit dem Kommando MODIFY-SYSTEM-PARAMETERS im laufenden Betrieb geändert werden:

| | | | |
|---|---|---|---|
| BLSCOPYN | EACTETYP | RDTESTPR | SSMSDEVB |
| BLSCOPYR | FILECRYP | SHUTARCH | SSMSDEVD |
| BLSLDPXS | FST32GB | SHUTCTL | SYSGJASL |
| BLSOPENX | ISBLKVAL | SHUTPROC | SYSGJCPU |
| DIATTL | JTABSMEM | SNAPTIME | SYSGJPRI |
| DUMPCL5P | JTMAXMEM | SSMAPRI | TCHOFLO |
| DUMPCTRL | JTSHMEM | SSMASEC | TCHREAD |
| DUMPSD# | JTSTDMEM | SSMCOPT | TCHTACK |
| DUMPSEPA | NBMSGCSD | SSMMILOG | VMGIORAL |
| DUMPSREF | NRTKILL | SSMOUT | WRTESTPR |

Diese Systemparameter werden in der folgenden Tabelle zusätzlich mit einem Stern (*) unter ihrem Namen markiert.

### Nicht-privilegierte Systemparameter

Folgende Systemparameter und ihre aktuellen Werte werden auch für nicht-privilegierte Anwender im Kommando SHOW-SYSTEM-PARAMETERS angezeigt:

| | | | |
|---|---|---|---|
| ASRSW1 | DMCMAXP | NBACODE | RDTESTPR |
| ASRSW2 | DUMPCL5P | NBESSIZE | SECSTART |
| AUDALLOW | DUMPSEPA | NBLOGENF | SECSTENF |
| BLKCTRL | ENCRYPT | NBMESLG | SHUTARCH |
| BLSCOPYN | FILECRYP | NBMSGCSD | SHUTPROC |
| BLSCOPYR | FMTYFNLG | NBOPTINT | SSMLGOF1 |
| BLSLDPXS | FREFCRYP | NBRCILU | SSMLGOF2 |
| CMDFILE | FST32GB | NBRCSCK | SSMOUT |
| DEFLUID | HOSTCODE | NBRCSCKN | SVC79 |
| DIATTL | MSGDEST | NBRUNSP | TEMPFILE |
| | | NBRUNWT | WRTESTPR |

Diese Systemparameter werden in der folgenden Tabelle zusätzlich mit einem Plus-Zeichen (+) unter ihrem Namen markiert.

## Systemparameter in alphabetischer Reihenfolge

| Name des Parameters | Bedeutung des Parameters | Standard-wert (Typ/Size) | Wahl-Wert |
|---|---|---|---|
| ASRSW1<br>+ | Nebenbedienstationen dürfen das ASR-Kommando mit den Operanden ADD,DELETE und PRIMARY ohne Angabe von Filtern ausführen, soweit sie den eigenen Status betreffen.<br><br>0 Ausführung für Nebenbedienstationen erlaubt.<br>1 Ausführung nur an der Hauptbedienstation erlaubt. | 0<br>(A/1) | 0/1 |
| ASRSW2<br>+ | Anwendungen, die als Bedienstation fungieren, haben bezüglich des ASR-Kommandos die gleichen Rechte wie:<br><br>0 eine Nebenbedienstation.<br>1 die Hauptbedienstation. | 0<br>(A/1) | 0/1 |
| AUDALLOW<br>+ | Legt fest, ob die Funktion AUDIT (Hardware- und Linkage-AUDIT) in der laufenden Session systemweit verfügbar sein soll.<br><br>YES AUDIT ist systemweit verfügbar<br>NO AUDIT ist nicht verfügbar. Ein bereits gestarteter Linkage-AUDIT wird ggf. beendet. | YES<br>(C/3) | YES/NO |
| BACKUP | Legt den Standard-BACKUP-Level für Dateien fest (für ARCHIVE-Sicherung). | A<br>(C/1) | A/B/C/<br>D/E |
| BLKCTRL<br>+ | Vergibt das Dateiattribut BLKCTRL entsprechend der Pam-key-Eigenschaft einer Platte (K- oder NK-Platte).<br><br>PAMKEY (kompatibler Modus)<br>Residiert die Datei auf einer K-Platte, erhält sie das Attribut BLKCTRL=PAMKEY.<br>Residiert die Datei auf NK-Platte, erhält sie das Attribut<br>– BLKCTRL=DATA für SAM- und ISAM-Dateien und<br>– BLKCTRL=NO für UPAM-Dateien.<br><br>NONKEY (Migrationsmodus)<br>Dieser Wert sollte gesetzt werden, wenn der Einsatz von NK-Platten/Pubsets geplant ist.<br><br>Unabhängig von der Platteneigenschaft werden die Dateien mit dem Attribut BLKCTRL=DATA/NO erzeugt. | PAMKEY<br>(C/6) | PAMKEY/<br>NONKEY |

Tabelle 55: Systemparameter (Teil 1 von 33)

| **Name des Parameters** | **Bedeutung des Parameters** | **Standard-wert (Typ/Size)** | **Wahl-Wert** |
|---|---|---|---|
| BLSCOPYN<br>+, * | Wert für den Copyright-Text, den der BINDER übernimmt, wenn in den Anweisungen START-LLM-CREATION und MODIFY-LLM-ATTRIBUTES beim Operanden COPYRIGHT der Wert SYSTEM-DEFAULT angegeben ist (siehe Handbuch „BINDER“ [6]).<br><br>name<br>alphanumerisch; darf maximal 63 Zeichen lang sein; Standardwert ist Blank, jedoch darf Blank nicht in name enthalten sein. | C'␣'<br>(C/63) | C'name' |
| BLSCOPYR<br>+, * | Wert für den Copyright-Text, den TSOSLNK übernimmt, wenn in der Anweisung PROGRAM beim Operanden COPYRIGHT kein Wert angegeben ist.<br><br>name<br>alphanumerisch; darf maximal 12 Zeichen lang sein; Standardwert ist Blank, jedoch darf Blank nicht in name enthalten sein. | C'␣'<br>(C/12) | C'name' |
| BLSLDPXS<br>+, * | Wert der Ladeadresse des Programms, den TSOSLNK übernimmt, wenn in den Anweisungen OVERLAY oder PROGRAM der Operand LOADPT=*XS angegeben ist. | X'10000000'<br>(X/4) | |
| BLSOPENX<br>* | Legt den OPEN-Modus für Programmbibliotheken und Elemente fest, die von den Makroaufrufen BIND und LINK eröffnet werden (siehe Handbuch „Bindelader-Starter“ [5]).<br><br>Y OPEN-Modus EXEC<br>N OPEN-Modus INPUT | Y<br>(C/1) | Y/N |
| BLSUSLIM | Legt die maximale Anzahl von Klasse-4-Speicherseiten fest, die den BLS-Metadaten gleichzeitig zugewiesen werden können.<br><br>Die BLS-Metadaten gehören zum Shared Code des Benutzers und werden mit dem ASHARE-Makro des DBL in den Common Memory Pool geladen. Wenn Wert=0 ist, kann der Shared Code des Benutzers über den ASHARE-Makro nicht geladen werden. | 100<br>(A/2) | 0...8192 |

Tabelle 55: Systemparameter (Teil 2 von 33)

| Name des Parameters | Bedeutung des Parameters | Standard-wert (Typ/Size) | Wahl-Wert |
|---|---|---|---|
| BMTNUM | Legt die Anzahl der Ein-/Ausgabe-Puffer für die Katalogverwaltung fest. Je höher der Wert, um so größer ist der Durchsatz der Katalogverwaltung.<br><br>Für jeden Pubset wird die angegebene Anzahl von Puffern angelegt. Die Puffer werden beim Importieren angelegt und beim Exportieren freigegeben.<br><br>Für alle Privatplatten gemeinsam wird mit BMTNUM die Anzahl der Puffer festgelegt. Der Wert darf nicht 0 sein, wenn Privatplatten unterstützt werden sollen.<br><br>Der Wert wird für einen Pubset nur dann gültig, wenn kein Wert in den Kommandos IMPORT-PUBSET oder ADD-MASTER-CATALOG-ENTRY festgelegt wurde. | 32<br>(A/1) | 0...255 |
| CATBUFR | Legt fest, ob der Ein-/Ausgabe-Puffer (Input/Output Buffer) der Katalogverwaltung im Klasse-3-Speicher (resident) sein soll oder nicht.<br><br>N nicht resident<br>Y resident<br><br>*Hinweis*<br>Wenn der Ein-/Ausgabe-Puffer resident ist, wird der Durchsatz der Katalogverwaltung erhöht. In diesem Fall werden jedoch 2048 * b Byte im Klasse-3- Speicher belegt (b=Wert von Systemparameter BMTNUM – siehe dort). | N<br>(C/1) | N/Y |
| CMDFILE<br>+ | Kommandodatei unmittelbar bei der Systemeinleitung automatisch abarbeiten.<br><br>Nein Leerzeichen<br>Ja dateiname: maximal 44 Zeichen | CMDFILE<br>(C/54) | datei--name /<br>Leer-zeichen |
| CONSDDE7 | Die Meldung `DMSODE7 SAM FILE CLOSED` soll über die Bedienstation und/oder SYSOUT ausgegeben werden.<br><br>Y Ausgabe über Bedienstation + SYSOUT<br>N Ausgabe über SYSOUT | N<br>(C/1) | N/Y |

Tabelle 55: Systemparameter (Teil 3 von 33)

| Name des Parameters | Bedeutung des Parameters | Standard-wert (Typ/Size) | Wahl-Wert |
|---|---|---|---|
| DEFLUID<br>+ | Legt die Standard-Benutzerkennung des Systems, die mit der Abkürzung $. angesprochen werden kann, fest.<br>Zusätzlich suchen die Kommandos START-EXECUTABLE-PROGRAM, CALL-PROCEDURE, ENTER-JOB und bestimmte Dienstprogramme unter dieser Benutzerkennung, falls der Benutzer keine Benutzerkennung angegeben hat, und die Datei nicht unter seiner eigenen Benutzerkennung katalogisiert ist (Secondary Read). | $TSOS<br>(C/15) | :catid:<br>$userid |
| DESTLEV | Legt fest, ob Plattenspeicherplatz beim Freigeben vom System überschrieben wird.<br>DESTLEV ist wirksam bei DELETE-FILE, MODIFY-FILE-ATTRIBUTES (mit SPACE= RELEASE), bei der F5-Label-Rekonstruktion und und beim Löschen von Elementen einer PLAM-Bibliothek.<br><br>0 Nur bei Dateien, die mit DESTROY-BY-DELETE=YES katalogisiert sind oder die mit OPTION=DESTROY-ALL gelöscht werden, werden die freigegebenen Extents mit Nullen überschrieben.<br><br>1 Alle Systemdateien zur Jobsteuerung (S.IN, SPOOL) werden mit DESTROY-BY-DELETE=YES eingerichtet.<br><br>4 Alle Extents werden vor der eigentlichen Freigabe mit Nullen überschrieben, unabhängig davon, mit welchem DESTROY-Wert die Datei eingerichtet ist bzw. gelöscht wird.<br><br>5 Zusätzlich zu dem Verhalten wie bei 4 wird die F5-Label-Rekonstruktion einbezogen. Um sicherzustellen, dass auch nach einem Systemabbruch keine ungelöschten freien Bereiche vorhanden sind, werden alle freien Seiten während der F5-Label-Rekonstruktion auf 0 gesetzt.<br><br>6 Zusätzlich zu dem Verhalten wie bei 5 werden die Extents, die einer Datei zugewiesen sind, beim logischen Erzeugen dieser Datei mit binären Nullen überschrieben.<br>D.h., alle PAM-Seiten dieser Datei werden zum OPEN OUTPUT/OUTIN-Zeitpunkt auf Null gesetzt. Falls die Datei mit Pamkeys behaftet ist, werden diese auf eine ungültige CFID gesetzt. | 0<br>(A/1) | 0/1/4/5/6 |

Tabelle 55: Systemparameter (Teil 4 von 33)

| Name des Parameters | Bedeutung des Parameters | Standard-wert (Typ/Size) | Wahl-Wert |
|---|---|---|---|
| DIATTL<br>+, * | Begrenzt die einer Dialog-Task zur Verfügung stehende Zeit.<br><br>Y Die Dialog-Task wird bei `Task-Time-Runout` beendet.<br>N Die Dialog-Task wird bei `Task-Time-Runout` nicht beendet.<br>Q Wie Y. Zusätzlich wird in allen Tasks, also auch Batch-Tasks, bei Erreichen des CPU-Limits die Meldung EXC0070 nicht mehr ausgegeben. Im Anschluss werden alle Zeitzuschläge (z.B. für die STXIT-Contingency zum Ereignis Time-Runout) auf eine Sekunde begrenzt. | N<br>(C/1) | N/Y/Q |
| DMCMAXP<br>+ | Legt die maximale Anzahl der Einträge im MRSCAT des Home-Pubsets fest. Ist diese Anzahl erreicht, können weitere Einträge nicht erzeugt werden, bevor nicht welche gelöscht werden.<br>Während des Startup kann der Maximalwert überschritten werden, da er ignoriert wird.<br>Werte, die keiner Zweierpotenz entsprechen, werden auf die nächste Zweierpotenz aufgerundet (128, 256, 512), mindestens aber 128. | 128<br>(A/4) | 1...4096 |
| DMMAXSC | Maximalwert (in PAM-Blöcken), bis zu dem die Sekundärzuweisung bei Dateierweiterung verdoppelt wird. Der Wert muss mindestens so groß sein wie DMSCALL.<br>Der Wert gilt für alle Privatplatten; für einen Pubset wird der Wert nur dann gültig, wenn er nicht pubset-spezifisch mit dem Kommando ADD-/MODIFY-MASTER-CATALOG festgelegt wurde. | 48<br>(A/4) | 3...65535 |
| DMPRALL | Primärzuweisung für Dateien in PAM-Blöcken als Standardwert für den SPACE-Operanden im Kommando CREATE-FILE.<br>Der Wert gilt für alle Privatplatten; für einen Pubset wird der Wert nur dann gültig, wenn er nicht pubset-spezifisch mit dem Kommando ADD-/MODIFY-MASTER-CATALOG festgelegt wurde. | 3<br>(A/4) | 3...65535 |

Tabelle 55: Systemparameter (Teil 5 von 33)

| Name des Parameters | Bedeutung des Parameters | Standard-wert (Typ/Size) | Wahl-Wert |
|---|---|---|---|
| DMSCALL | Sekundärzuweisung für Dateien in PAM-Blöcken als Standardwert für SECONDARY-ALLOCATION im SPACE-Operanden des Kommandos CREATE-FILE; bei der ersten Erweiterung wird die durch DMSCALL bestimmte Anzahl von PAM-Blöcken zugewiesen, bei jeder weiteren Erweiterung wird die Zuweisung verdoppelt bis maximal zum Wert von DMMAXSC.<br>Der Wert gilt für alle Privatplatten; für einen Pubset wird der Wert nur dann gültig, wenn er nicht pubset-spezifisch mit dem Kommando ADD-/MODIFY-MASTER-CATALOG festgelegt wurde. | 3<br>(A/4) | 3...65535 |
| DUMPCL5P<br>+, * | Steuert in CDUMP, ob der privilegierte Klasse-5-Speicher im Userdump oder Areadump enthalten sein soll.<br><br>0 Der gesamte Klasse-5-Speicher ist im Dump enthalten.<br>1 Der privilegierte Klasse-5-Speicher wird bei der Ausgabe von User- und Areadumps unterdrückt. | X'00'<br>(X/1) | X'00'/<br>X'01' |

Tabelle 55: Systemparameter (Teil 6 von 33)

| Name des Parameters | Bedeutung des Parameters | Standard-wert (Typ/Size) | Wahl-Wert |
|---|---|---|---|
| DUMPCTRL<br>* | Schaltet dynamische Dump-Eigenschaften ein. Folgende Einstellwerte sind möglich:<br><br>Bit $2^0$ gesetzt:<br>Duplikatserkennung eingeschaltet; Duplikate werden unterdrückt. Anstelle des Dump wird die Meldung `IDA0N56` ausgegeben.<br>Bit $2^1$ gesetzt:<br>Operatorloser Betrieb eingeschaltet; die Meldung `IDA0N52` wird unterdrückt. Dieser Wert hat auf User- und Areadumps keine Auswirkung.<br>Bit $2^2$ gesetzt:<br>Die Meldung `IDA0N52` wird unterdrückt, wenn sich der Fehler, der den Dump verursacht hat, auf eine System-task bezieht.<br>Bit $2^3$ gesetzt:<br>Im Fall eines abnormalen Dumpabbruchs wird bei Systemdumps die Meldung `IDA0N99` ausgegeben.<br>Bit $2^4$ gesetzt:<br>Beim FILE-Aufruf für die Ausgabedatei des Dump wer-den die Parameter IOPERF=HIGH und IOUSAGE= WRITE nicht verwendet. Damit wird bei Einsatz von SMS die Dump-Datei immer auf der Standard Storage-Class abgelegt.<br>Dieser Wert ist nur bei Einsatz von SMS sinnvoll.<br>Ist DUMPCTRL=X'00', sind alle oben beschriebenen Funk-tionen ausgeschaltet. | X'00'<br>(X/1) | X'00'/<br>X'0F' |
| DUMPSD#<br>* | Legt die Anzahl der Systemdumps fest, die pro Session „automatisch“ (ohne Operator-Dialog) ausgegeben werden sollen.<br>Der Parameter hat auf User- und Areadumps keinen Ein-fluss.<br><br>X'00' Bereits beim ersten Systemdump erfolgt ein Operator-Dialog. | X'00'<br>(X/1) | X'00'/<br>X'FF' |

Tabelle 55: Systemparameter (Teil 7 von 33)

| **Name des Parameters** | **Bedeutung des Parameters** | **Standard-wert (Typ/Size)** | **Wahl-Wert** |
|---|---|---|---|
| DUMPSEPA<br>+, * | Legt die Ausgabe von Secret Pages bei User- und System-dumps fest.<br><br>1 Die Ausgabe von Secret Pages wird nicht unterdrückt.<br>2 Die Ausgabe von Secret Pages des Klasse-6-Speichers wird unterdrückt. Die Ausgaben von Secret Pages des Klasse-1/2/3/4/5-Speichers werden nicht unterdrückt.<br>3 Die Ausgabe von Secret Pages des Klasse-1/2/3/4/5-Speichers wird unterdrückt. Die Ausgabe von Secret Pages des Klasse-6-Speichers wird nicht unterdrückt.<br>4 Die Ausgabe von allen Secret Pages wird unterdrückt. | 4<br>(A/4) | 1/2/3/4 |
| DUMPSREF<br>* | Steuert die Ausgabe der Klasse-6-Speicherseiten beim Systemdump.<br><br>0 Ausgabe der referenzierten Klasse-6-Speicherseiten<br>1 Ausgabe aller Klasse-6-Speicherseiten<br>2 keine Ausgabe der Klasse-6-Speicherseiten<br>3 Ausgabe der Klasse-6-Speicherseiten ohne schreib-geschützte Seiten | X'00'<br>(X/1) | |
| EACTETYP<br>* | Legt fest, welche der folgenden Meldungen über die Bedienstation ausgegeben werden sollen: `BLS0500`, `BLS0517`, `BLS0519`, `BLS0524`, `BLS0526`, `BLS0539`, `BLS0551` und `BLS0552`<br><br>0 keine der Meldungen<br>1 nur die Meldung `BLS0519`<br>2 alle oben genannten Meldungen<br>3 alle Meldungen außer `BLS0519` | 0<br>(A/1) | 0/1/2/3 |
| EAMMEM | Legt die Größe des Klasse-4-Speichers, der für EAM ge-nutzt wird, fest. Dabei werden bei EAM-Aufrufen statt Ein-/Ausgaben auf die Platte nur MVCs durchgeführt.<br><br>Dieser Parameter gilt nur für den Home-Pubset.<br>Hinweise für sinnvolle Größenangaben: siehe Seite 423 | 0<br>(A/2) | 0...2730 |

Tabelle 55: Systemparameter (Teil 8 von 33)

| Name des Parameters | Bedeutung des Parameters | Standard-wert (Typ/Size) | Wahl-Wert |
|---|---|---|---|
| EAMMIN | Legt die Minimalgröße der Datei SYSEAM fest (in Units). Wird beim ersten EAM-Zugriff festgestellt, dass die SYSEAM-Datei kleiner als EAMMIN ist, wird sie auf diese Größe erweitert bzw. in dieser Größe angelegt.<br><br>Bei erweiterter Unterstützung von EAM auf Shared-Pubsets (siehe EAMSPVS) arbeitet jedes System, das einen Shared-Pubset importiert, mit seinem eigenen Wert von EAMMIN.<br>Hinweise für sinnvolle Größenangaben: siehe Seite 422 | 3000<br>(A/2) | 4...64512 |
| EAMSEC | Legt die Sekundärzuweisung der Datei SYSEAM fest (in Units). Der Wert sollte ein Vielfaches von 8 betragen.<br><br>Bei erweiterter Unterstützung von EAM auf Shared-Pubsets (siehe EAMSPVS) arbeitet jedes System, das einen Shared-Pubset importiert, mit seinem eigenen Wert von EAMSEC.<br>Hinweise für sinnvolle Größenangaben: siehe Seite 422 | 200<br>(A/2) | 1...64512 |
| EAMSIZ | Legt die Größe von SYSEAM fest (in Units), die einem Benutzer allein zur Verfügung steht.<br>Bei erweiterter Unterstützung von EAM auf Shared-Pubsets (siehe EAMSPVS) arbeitet jedes System, das einen Shared-Pubset importiert, mit seinem eigenen Wert von EAMSIZ.<br>Hinweise für sinnvolle Größenangaben: siehe Seite 423 | 64512<br>(A/2) | 4...64512 |
| EAMSPVS | Steuert die Behandlung der SYSEAM-Datei auf einem Shared-Pubset (SPVS).<br><br>X'00' Nur der Master eines SPVS kann eine SYSEAM-Datei auf diesem SPVS einrichten (Dateiname: :<SPVS-id>:$TSOS.SYSEAM)<br>X'01' Jedes System mit Zugriff auf den SPVS kann auf ihm eine SYSEAM-Datei einrichten (Dateiname: :<SPVS-id>:$TSOS.SYSEAM.<sysid>) | X'00'<br>(X/1) | X'00'/<br>X'01' |
| ENCRYPT<br>+ | Verschlüsselung von Datei- und Benutzerkennwörtern erwünscht?<br><br>Y Kennwörter werden verschlüsselt abgelegt.<br>N Kennwörter werden unverschlüsselt abgelegt. | N<br>(C/1) | N/Y |

Tabelle 55: Systemparameter (Teil 9 von 33)

| **Name des Parameters** | **Bedeutung des Parameters** | **Standard-wert (Typ/Size)** | **Wahl-Wert** |
|---|---|---|---|
| ERECRYPT | Eingabeparameter für CHECKPOINT/RESTART zur Ermittlung einer Kontrollzahl für den Checkpoint.<br>Wert:<br><br>0 für den Checkpoint wird keine Kontrollzahl ermittelt<br>aaaabbbbcccccccc<br>Parameter, die in die Berechnung der Kontrollzahl einfließen; jeder Parameter sollte ungleich 0 sein | 0<br>(X/8) | X 'aaaa bbbb cccc cccc' |
| EREPASSW | Steuert die Checkpoint/RESTART-Funktionalität.<br><br>N Die gesamte Checkpoint/Restart-Funktionalität ist zulässig, die Checkpoint-Dateien werden nicht mit einem Zufalls-Kennwort geschützt.<br>Y Checkpoints können nur noch mit dem Makro WRCPT erstellt werden. CHKPT ist nur noch dann zulässig, wenn der zugehörige FCB zu einer *DUMMY-Datei gehört.<br>Das Schreiben von Checkpoints am Bandende bei VLTFs (Very Large Tape Files) ist nicht möglich.<br>Zusätzlich werden die Checkpoint-Dateien mit einem Zufalls-Kennwort geschützt (ähnlich wie S.IN.-Dateien). | N<br>(C/1) | N/Y |
| ETMFXLOW | Tasks mit einer externen Priorität schlechter (d.h. größer) oder gleich ETMFXLOW erhalten in TU keinen Alterungszuschlag. Ein evtl. in TPR gewonnener Prioritätszuwachs verfällt beim Übergang nach TU. | 256<br>(A/2) | 127...256 |
| FARMTSAV | Legt fest, dass eine Datei bei der Differenzsicherung mit ARCHIVE gesichert wird, wenn der Dateiinhalt oder der Katalogeintrag der Datei geändert wird.<br><br>0 Die Datei wird gesichert, wenn ihr Inhalt geändert wurde.<br>1 Die Datei wird auch gesichert, wenn nur ihr Katalogeintrag geändert wurde. | 0<br>(X/1) | 0/1 |

Tabelle 55: Systemparameter (Teil 10 von 33)

| Name des Parameters | Bedeutung des Parameters | Standard-wert (Typ/Size) | Wahl-Wert |
|---|---|---|---|
| FILECRYP<br>+, * | Bestimmt das Verschlüsselungsverfahren beim Umwandeln in eine verschlüsselte Datei.<br>Es werden die Verschlüsselungsverfahren AES (Voreinstellung) und DES unterstützt.<br><br>Der aktuelle Wert des Systemparameters FILECRYP wird beim Kommando ENCRYPT-FILE in die Verschlüsselungsmerkmale der Datei übernommen.<br>Beim Zugriff auf eine bereits verschlüsselte Datei wird das Verschlüsselungsverfahren dem Katalogeintrag der Datei entnommen.<br>Eine Änderung des Systemparameters FILECRYP wird für eine zum Zeitpunkt der Änderung bereits verschlüsselte Datei erst dann wirksam, wenn die Datei entschlüsselt und anschließend erneut verschlüsselt wird.<br><br>Näheres zu den Verschlüsselungsverfahren steht im Benutzerhandbuch zu „openCRYPT“ [9]. | AES<br>(C/4) | AES/DES |
| FMTYFNLG<br>+ | Legt fest, wie der Dateiname verschiedener Logging-Dateien (CONSLOG, ACCOUNT) bzgl. der Datumsangabe gebildet wird.<br><br>2 Der Dateiname wird mit 2-stelligen Jahresangaben gebildet: <präfix>.jj.mm.tt.<suffix><br>4 Der Dateiname wird mit 4-stelligen Jahresangaben gebildet: <präfix>.jjjj-mm-tt.<suffix> | 4<br>(C/1) | 2/4 |
| FREFCRYP<br>+ | Enthält - falls nicht leer - eine ausgewählte Benutzerkennung. Nur für Dateien von dieser Kennung ist dann die Umwandlung in verschlüsselte Dateien (Kommando ENCRYPT-FILE) mit freier Definition eines Crypto-Kennworts möglich.<br><br>Bei der Umwandlung von Dateien anderer Benutzerkennungen muss dann eine schon verschlüsselte Referenzdatei angegeben werden.<br><br>Die Menge aller verwendeten Crypto-Kennwörter wird damit begrenzt auf die der verschlüsselten Dateien von der ausgewählten Benutzerkennung. | C'␣'<br>(C/8) | C'1...8' |

Tabelle 55: Systemparameter (Teil 11 von 33)

| Name des Parameters | Bedeutung des Parameters | Standard-wert (Typ/Size) | Wahl-Wert |
|---|---|---|---|
| FSHARING | Legt fest, ob der im Benutzerkatalog definierte Zugriffs-schutz zu einem Pubset wirksam ist oder nicht.<br><br>*Zugriff auf einen Pubset am lokalen System*<br>0 Nur diejenigen Benutzer können auf einen Pubset zugreifen, die einen Eintrag im Benutzerkatalog in diesem Pubset haben.<br>1 Jeder Benutzer kann auf jeden Pubset zugreifen.<br><br>*Zugriff auf einen Pubset in einem entfernten System*<br>0 Nur diejenigen Benutzer können auf einen Pubset am entfernten System zugreifen, die sowohl einen Eintrag im Benutzerkatalog in diesem Pubset haben als auch im Benutzerkatalog des Home-Pubsets des entfernten Systems.<br>1 Jeder Benutzer kann auf jeden Pubset zugreifen.<br><br>Dies gilt für den Standardfall einer Rechner-Verbindung, den CCS-Verbund (Closely Coupled System).<br><br>Aus Kompatibilitätsgründen wird auch weiterhin der LCS-Verbund (Loosely Coupled System) unterstützt. FSHARING ist im LCS-Verbund ohne Bedeutung, d.h. er wird nicht aus-gewertet. Es gelten – abhängig von der Pubset-Art – folgen-de Zugriffsbedingungen:<br><br>*Zugriff auf einen entfernten SF-Pubset*<br>Nur diejenigen Benutzer können auf einen Pubset am ent-fernten System zugreifen, die einen Eintrag im Benutzerka-talog in diesem Pubset haben, wobei das Kennwort in die-sem Eintrag mit dem Logon-Kennwort am Home-Pubset des lokalen Systems übereinstimmen muss.<br><br>*Zugriff auf einen entfernten SM-Pubset*<br>Nur diejenigen Benutzer können zugreifen, die sowohl einen Eintrag im Benutzerkatalog in diesem Pubset haben, als auch einen Eintrag im Benutzerkatalog auf dem Home-Pub-set des entfernten Systems, wobei das Kennwort im Home-Pubset des entfernten Systems mit dem Logon-Kennwort am Home-Pubset des lokalen Systems übereinstimmen muss. | 0<br>(A/1) | 0/1 |

Tabelle 55: Systemparameter (Teil 12 von 33)

| **Name des Parameters** | **Bedeutung des Parameters** | **Standard-wert (Typ/Size)** | **Wahl-Wert** |
|---|---|---|---|
| FST32GB<br>* | Legt den Umgang mit Dateien ≥ 32 GB fest.<br><br>0 Alle FSTAT-Schnittstellen mit Version=0 oder Version=1 (≙ Version=710 bzw. Version=800), die nicht den Operanden FNAM spezifiziert haben, werden mit dem Returncode 0576 zurückgewiesen, wenn sich in der Treffermenge eine Datei ≥ 32 GB befindet.<br>1 (3-Byte-Datenfeld-Überlauf ignorieren)<br>Befindet sich in der Treffermenge eine Datei ≥ 32 GB, wird kein Fehler angezeigt. Den nicht darstellbaren Datenfeldern wird der Wert X'FFFFFF' zugewiesen. | 0 | 0 / 1 |
| HOSTCODE<br>+ | Legt den „Codierten Zeichensatz“ fest, der als System-Standard-Zeichensatz verwendet werden soll.<br><br>name<br>maximal acht Zeichen langer Name des codierten Zeichensatzes. | EDF03IRV<br>(C/8) | name |
| ISBLKVAL<br>* | Steuert die Validierung von Blöcken einer NK-ISAM-Datei bei lesendem Zugriff.<br><br>X'00' Es findet keine Validierung statt.<br>X'01' Es findet eine Validierung statt. | X'00'<br>(X/1) | X'00'/<br>X'01' |
| ISPLDEFC | Legt das von ISAM verwendete Pubset fest, wenn bei der Bearbeitung von ISAM-Pools keine Katalogkennung angegeben wird.<br><br>X'00'<br>ISAM wählt den im Benutzerkatalog-Eintrag des Benutzers angegebenen DEFAULT-PUBSET.<br><br>X'01'<br>ISAM wählt den Home-Pubset. | X'00'<br>(X/1) | X'00'/<br>X'01' |
| JTABSMEM<br>* | Legt auf SQ-Servern (nicht aber SQ100) den maximalen CISCFW-Speicherplatz - aufsummiert über alle laufenden Programme - auf dem gesamten Server fest (in MB).<br><br>Beim Standardwert 0 berechnet sich das Subsystem JITSYS den Wert selbst (JIT = Just-in-time). | 0<br>(A/4) | 0...<br>1000000 |

Tabelle 55: Systemparameter (Teil 13 von 33)

| Name des Parameters | Bedeutung des Parameters | Standard-wert (Typ/Size) | Wahl-Wert |
|---|---|---|---|
| JTMAXMEM * | Legt auf SQ-Servern (nicht aber SQ100) den Maximalwert für die Größe der CISCFW-Workarea pro Task fest (in MB). Das ist der Wert, der im Kommando MODIFY-DBL-DEFAULTS *CMD-CALL(CISC-COMPILATION=YES(WORKSPACE=n)) als Maximalwert des Parameters WORKSPACE möglich ist.<br>*Hinweis*<br>CISCFW holt jeweils nur so viel Speicher, wie zum Ablauf des aktuellen /390-Programms notwendig ist. Bei JTMAXMEM=0 laufen die Programme nur im langsamen Interpretermodus ab. | 128 | 0...65535 |
| JTSHMEM * | Legt auf SQ-Servern (nicht aber SQ100) fest, wieviel Speicherplatz JITSYS zur Ablage von Share-Kompilaten verwenden soll (in MB).<br><br>Share-Kompilate entstehen bei der Emulation von in Klasse-4-Speicher geladenen Subsystemen. Sind keine solchen Subsysteme vorhanden bzw. sollen diese nicht „shared“ übersetzt werden, kann mit JTSHMEM = 0 die Erzeugung von „shared“ Kompilaten verhindert werden.<br>Der Wert JTSHMEM sollte in Abstimmung mit dem insgesamt zur Verfügung stehenden Klasse-3-Speicher gesetzt werden.<br><br>Die bei JTSHMEM angegebene Speichergröße wird bei Initialisierung von JITSYS sofort allokiert. Eine Änderung des Werts im laufenden Betrieb wird bis auf weiteres nur unterstützt, wenn der Wert vergrößert wird. Die Änderung des Wertes kann nur in 4er-Schritten erfolgen, andere Werte werden auf die nächste 4-MB-Grenze aufgerundet. | 64 (A/2) | 0...256 |
| JTSTDMEM * | Legt auf SQ-Servern (nicht aber SQ100) den Standardwert für die Größe der CISCFW-Workarea pro Task fest (in MB). Das ist der Wert, der im Kommando MODIFY-DBL-DEFAULTS *CMD-CALL(CISC-COMPILATION=YES(WORKSPACE=*STD)) als Standardwert des Parameters WORKSPACE verwendet wird.<br><br>Der Parameter kann nur in 4-MB-Schritten vergeben werden. JITSYS rundet auf die nächste 4-MB-Grenze auf. JITSYS holt jeweils nur soviel Speicher, wie zum Ablauf des aktuellen /390-Programms notwendig ist. | 16 | 0...65535 |

Tabelle 55: Systemparameter (Teil 14 von 33)

| Name des Parameters | Bedeutung des Parameters | Standard-wert (Typ/Size) | Wahl-Wert |
|---|---|---|---|
| L4MSG | Steuert die Ausgabe der Meldung<br>`EXC044E: 'ACCEPT ALLOCATION-REQUEST...'`<br><br>0 Meldung wird nicht ausgegeben<br>1 Meldung wird ausgegeben<br>*Ausnahme*<br>Die Anforderung kommt vom Slave eines Shared Pubsets. | 0<br>(A/1) | 0/1 |
| L4SPDEF | Voreinstellung für die Sättigungsstufe 4 des Pubset-Plattenspeicherplatzes. Eine Sättigungsstufe 4 für einen Pubset ist erreicht, wenn weniger freie PAM-Blöcke verfügbar sind als im Parameter angegeben.<br>n kann bis zu 50.000.000 groß sein.<br><br>Für einen Pubset wird der Wert nur dann gültig, wenn er nicht pubset-spezifisch mit dem Kommando ADD-/MODIFY-MASTER-CATALOG-ENTRY festgelegt wurde. | 2500<br>(A/4) | 66...n |

Tabelle 55: Systemparameter (Teil 15 von 33)

| Name des Parameters | Bedeutung des Parameters | Standard-wert (Typ/Size) | Wahl-Wert |
|---|---|---|---|
| MCXSPXCS | Bestimmt, ob der Rechner XCS-fähig ist oder nicht.<br><br>N der Rechner ist nicht XCS-fähig<br>In der MSCF-Konfigurationsdatei darf kein XCS-Name angegeben sein. Der Rechner wird ein CCS-Rechner und nur die SPVS-Funktionaltät ist verfügbar.<br>V der Rechner ist XCS-fähig<br>Abhängig von der Angabe des XCS-Namens in der MSCF-Konfigurationsdatei wird der Rechner ein XCS-Rechner im XCS-Verbund (er richtet diesen Verbund ein oder tritt ihm bei) oder ein CCS-Rechner (nur SPVS-Funktionalität). Im CCS-Modus werden Anforderungen von XCS-globalen Ressourcen abgewiesen und die XCS-Funktionalität ist nicht verfügbar, jedoch kann MSCF später im XCS-Modus gestartet werden.<br>Y der Rechner ist XCS-fähig<br>Abhängig von der Angabe des XCS-Namens in der MSCF-Konfigurationsdatei wird der Rechner ein XCS-Rechner im XCS-Verbund (er richtet diesen Verbund ein oder tritt ihm bei) oder ein CCS-Rechner (nur SPVS-Funktionalität). Im CCS-Modus werden Anforderungen von XCS-globalen Ressourcen durch systemglobale Ressourcen befriedigt. Dadurch steht die XCS-Funktionalität system-lokal zur Verfügung, doch ein Start von MSCF im XCS-Modus ist in diesem Systemlauf nicht mehr möglich. | N<br>(C/1) | N/V/Y |
| MSGCENTL | Legt die Länge eines Eintrags im Klasse-4-Speicher für die Meldungsbearbeitung fest (in Byte). Der Wert muss ein Vielfaches von 4 sein. In diesem Bereich des Klasse-4-Speichers werden die zuletzt und am häufigsten benutzten Meldungen zwischengespeichert, um Dateizugriffe zu sparen. | 200<br>(A/2) | 36...2500 |
| MSGCENTN | Legt die Anzahl von Einträgen im Klasse-4-Speicher (siehe MSGCENTL) für die Meldungsbearbeitung fest.<br><br>*Hinweis*<br>Der Bedarf an Klasse-4-Speicher kann durch Multiplizieren der Werte von MSGCENTL und MSGCENTN berechnet werden. | 32<br>(A/2) | 0...32767 |

Tabelle 55: Systemparameter (Teil 16 von 33)

| Name des Parameters | Bedeutung des Parameters | Standard-wert (Typ/Size) | Wahl-Wert |
|---|---|---|---|
| MSGDEST<br>+ | Bei Ausgabe von Texten über das Kommando INFORM-OPERATOR oder über Makro (MSG7, TYPIO, ...) ohne Zielangabe wird der hier angegebene Routing-Code (x), der berechtigte Benutzerprozess (nnnn) oder die Bedienstation (mn) als Ziel angenommen.<br>Werte: siehe Parameterservice-Anweisung SET-CODE<br>Bei Angabe eines nicht generierten berechtigten Benutzerprozesses oder einer nicht generierten Bedienstation wird beim Laden des Systems der Standardwert angenommen. | '<*␣␣'<br>(C/4) | '<x␣␣'/<br>'nnnn'/<br>'(mn)' |
| MSGDLAM | Anzahl der Meldungsdateien, die über DLAM verarbeitet werden sollen. | 0<br>(A/1) | 0...99 |
| MSGFIL01<br>MSGFIL02<br>MSGFIL03<br>MSGFIL04<br>.<br>.<br>.<br>.<br>MSGFIL15 | Anzahl und Dateinamen der Meldungsdateien.<br>Es gelten folgende Standardeinstellungen:<br>MSGFIL01=SYSMES.BS2CP.180<br>MSGFIL02=SYSMES.EKP.01<br><br>Für die Parameter MSGFIL03 bis MSGFIL15 gibt es keine Standardnamen, d.h. diese Dateien werden nicht automatisch eingerichtet.<br><br>dateiname muss vollqualifiziert sein und kann nur mit der Benutzerkennung $TSOS katalogisiert werden.<br>Wenn $TSOS nicht angegeben ist, darf dateiname maximal 41 Zeichen lang sein. | (s. links)<br>(C/51) | datei-name(n) |
| MSGLPRI | Sprache der Meldungsausgabe.<br><br>ED Englisch, Deutsch<br>DE Deutsch, Englisch<br><br>Standard = Englisch; ist kein englischer Eintrag vorhanden, so wird deutscher Meldungstext ausgegeben (Priorität).<br>Weitere Sprachen sind möglich, falls sie in die Meldungsdateien eingetragen sind. | ED<br>(C/2) | ED/DE |
| MSGNOFL | Anzahl der Meldungsdateien, die durch die Parameter MSGFILxx spezifiziert werden (siehe oben).<br><br>=0 Mindestens ein Name für eine Meldungsdatei wird über die Bedienstation angefordert.<br>>0 Dateinamen werden über die Bedienstation angefordert, wenn der Dateiname im Parameter MSGFILxx ungültig ist. | 2<br>(A/1) | 0...15 |

Tabelle 55: Systemparameter (Teil 17 von 33)

| Name des Parameters | Bedeutung des Parameters | Standard-wert (Typ/Size) | Wahl-Wert |
|---|---|---|---|
| NBACODE<br>+ | Legt den Standard-Berechtigungsschlüssel beim Eintragen neuer Operator-Kommandos fest (dynamische Bestellung). | E<br>(C/1) | A-Z / 0-9 / $ / * / @/ # |
| NBBAPRIV | Es kann ein verschärfter Zugangsschutz zu $CONSOLE-Anwendungen eingeschaltet werden.<br><br>N Beliebige Applikationen können sich an $CONSOLE anschließen, sofern ein evtl. spezifiziertes Kennwort mit gegeben wird. Lokale Applikationen, die mit einem $-Zeichen beginnen, können sich auch ohne Kennwort an $CONSOLE anschließen.<br>Y Es dürfen sich nur lokale Applikationen an $CONSOLE anschließen, die mit einem $-Zeichen beginnen (diese dürfen nur unter TSOS eröffnet werden). D.h., die Konnektierung wird abgewiesen, falls der Prozessorname der Applikation nicht der Name des Home-Prozessors ist, oder falls die Applikation nicht mit einem $-Zeichen beginnt. | N<br>(C/1) | N/Y |
| NBCONOPI | Stellt den Modus für die Bedienung an physikalischen Konsolen und an $CONSOLE-Anwendungen mit dynamischen Berechtigungsnamen ein.<br><br>Y Operator-Logon ist eingeschaltet<br>N Operator-Logon ist ausgeschaltet | N<br>(C/1) | Y/N |
| NBESSIZE<br>+ | Legt die max. Größe der zentralen System-Ereignisstrom-Datei $SYSAUDIT.SYSLOG.ESS.SYSTEM in PAM-Seiten fest. Bei Erreichen der angegebenen Maximalgröße werden die ältesten protokollierten Ereignisse gelöscht und der gewonnene Speicherplatz für die neu aufzuzeichnenden Daten verwendet.<br>Die Feinauflösung der eingestellten Werte ist abhängig von der Sekundärzuweisung für diese Datei.<br>Bei Größe 0 wird die Datei nicht angelegt. Die System-Ereignisstromdienste sind nicht verfügbar. | 40000<br>(A/4) | 0 /<br>10.000 ...<br>200.000 |

Tabelle 55: Systemparameter (Teil 18 von 33)

| **Name des Parameters** | **Bedeutung des Parameters** | **Standard-wert (Typ/Size)** | **Wahl-Wert** |
|---|---|---|---|
| NBIMPRCA | Bestimmt, ob die impliziten Berechtigungen nur bis „System Ready“ und nach SHUTDOWN END wirksam sein sollen oder ob sie während des gesamten Systemlaufs verwaltet, d.h. der jeweiligen Hauptkonsole zugeordnet werden.<br><br>Y Die impliziten Berechtigungen wirken während des gesamten Systemlaufs.<br>N Die impliziten Berechtigungen sind an der Hauptkonsole nur bis „System Ready“ und nach SHUTDOWN END wirksam.<br><br>Dieser Systemparameter wird bei NBCONOPI=N nicht ausgewertet. | N<br>(C/1) | Y/N |
| NBKESNR | Steuert die Namensbildung und die Kennung, auf die die SYS.CONSLOG-Dateien geschrieben werden.<br>NBKESNR=3 führt z.B. zum vollqualifizierten Dateinamen $TSOS.SYS.CONSLOG.yyyy-mm-tt.sss.nn<br>Fehlende Angabe oder NBKESNR=0 führt zum Dateinamen $SYSAUDIT.SYS.CONSLOG.yyyy-mm-tt.sss.nnn<br><br>0 SYSAUDIT-Kennung und 3-stellige Seriennummer<br>1 TSOS-Kennung und 3-stellige Seriennummer<br>2 SYSAUDIT-Kennung und 2-stellige Seriennummer<br>3 TSOS-Kennung und 2-stellige Seriennummer | 0<br>(A/1) | 0/1/2/3 |
| NBLDADMP | Legt fest, ob die Operanden des DADM-Kommandos, in denen Kennwörter enthalten sein können, in der CONSLOG-Datei protokolliert werden.<br><br>F full logging<br>E erase operands | F<br>(C/1) | E/F |

Tabelle 55: Systemparameter (Teil 19 von 33)

| Name des Parameters | Bedeutung des Parameters | Standard-wert (Typ/Size) | Wahl-Wert |
|---|---|---|---|
| NBLOGENF<br>+ | Legt fest, ob CONSLOG nach nicht behebbarem Fehler DMS054D mit dem Kommando CHANGE-CONSLOG wieder eingeschaltet werden kann.<br><br>E Bei Eröffnen der SYS.CONSLOG wird eine zusätzliche Datei als Platzhalter angelegt. Ist dies nicht möglich, wird der Operator gewarnt. Kommt es trotzdem zur Meldung DMS054D, wird der Operator mit der Meldung NBR0953 aufgefordert, das Console Logging fortzusetzen, nachdem Platz geschaffen wurde (Antwort „R“), oder den Systemlauf abzubrechen (Antwort „H“).<br>N Nach der Meldung DMS054D wird die Meldung EXC0991 CONSLOG INACTIVE ausgegeben. Nach Behebung der Speicherplatzprobleme kann mit CHANGE-CONSLOG eine neue Logging-Datei eröffnet werden. Die Seriennummer der SYS.CONSLOG wird um zwei erhöht, um den Ausfall zu protokollieren.<br><br>Wenn die durch NBKESNR festgelegte max. CONSLOG-Seriennummer bereits erreicht ist, wird bei NBLOGENF=E das Kommando CHANGE-CONSLOG abgewiesen, bei NBLOGENF=N die aktuelle Logging-Datei geschlossen und das Console Logging abgeschaltet. | E<br>(C/1) | E/N |
| NBMESLG<br>+ | Legt fest, ob die Nachrichten von INFORM-JOB oder INFORM-ALL-JOBS in ihrer vollen Länge (L) oder in maximal einer Zeile (S) ausgegeben werden sollen. | L<br>(C/1) | S/L |
| NBMSGCSD<br>+, * | Gibt an, ob im Zustand „alle (physikalischen) Konsolen defekt“ anfallende Systemmeldungen als ausgegeben betrachtet werden sollen, obwohl sie an keiner physikalischen Konsole ausgegeben wurden.<br><br>N Alle Meldungen, für die eine physikalische Konsole berechtigter Empfänger ist, werden solange im Systemspeicher gepuffert, bis die Ausgabe auf eine physikalische Konsole möglich ist.<br>Y Nach der Verteilung an andere Empfänger (z.B. CONSLOG) werden die Meldungen freigegeben. Im System entsteht dadurch kein Speicherplatzmangel. | N | N/Y |

Tabelle 55: Systemparameter (Teil 20 von 33)

| Name des Parameters | Bedeutung des Parameters | Standard-wert (Typ/Size) | Wahl-Wert |
|---|---|---|---|
| NBOPTINT<br>+ | Zeitintervall in Minuten, nach dem auf allen Bedienstationen die Meldung NBR0750 ausgegeben wird, die das Tagesdatum enthält. Der Routing-Code dieser Meldung ist „E“.<br><br>0 Die Meldung NBR0750 wird nicht ausgegeben. | 0<br>(A/1) | 0 ... 255 |
| NBRCILU<br>+ | Gibt die Zeiteinheit in Millisekunden an, mit der das Zeitintervall der Kommandos (CLI) multipliziert wird, um die effektive Operator-Task-Belegungszeit festzulegen.<br><br>Bei zu häufigem Auftreten der Meldung NBR0974 sollte diese Einheit erhöht werden. | 5000<br>(A/4) | 1000 ...<br>120000 |
| NBRCSCK<br>+ | Legt die Zeitdauer in Minuten fest, nach der die Verbindung eines privilegierten, berechtigten Benutzerprozesses (TSOS-Anwendung des eigenen Systems) an die UCON-Task ($CONSOLE) abgebrochen wird, wenn seinerseits keine Nachrichten mehr abgeholt werden. | 10<br>(X/1) | 1 ... 255 |
| NBRCSCKN<br>+ | Legt die Zeitdauer in Minuten fest, nach der die Verbindung eines nichtprivilegierten, berechtigten Benutzerprozesses (nicht TSOS-Anwendung des eigenen Systems) an die UCON-Task ($CONSOLE) abgebrochen wird, wenn seinerseits keine Nachrichten mehr abgeholt werden. | 10<br>(X/1) | 1 ... 255 |
| NBREPLY | Legt fest, ob mit dem Berechtigungsschlüssel * die Berechtigung zur Beantwortung aller im System offenen Operator-(normal-) Fragen verknüpft sein soll.<br><br>Ausnahme: Emergency-Fragen und Kommandozusatzinformationsanforderungen von Benutzertasks mit dem Privileg OPERATING.<br><br>N Die Berechtigung soll nicht mit dem BS * verknüpft sein.<br>Y Die Berechtigung soll mit dem BS * verknüpft sein. | N<br>(C/1) | N/Y |
| NBRUNSP<br>+ | Legt fest, ob Kommandos aus RUN-Sequenzen an der Bedienstation ausgegeben werden sollen, von der das RUN-Kommando eingegeben wurde. | N<br>(C/1) | N/Y |
| NBRUNUID | Gibt an, unter welcher Benutzerkennung eine RUN-Datei (CMDFILE) gesucht werden soll, falls keine explizite Angabe erfolgte.<br>Ist das erste Zeichen kein „$“, wird die Angabe ignoriert. Es wird unter der Benutzerkennung TSOS gesucht. | C' '<br>(C/9) | $userid |

Tabelle 55: Systemparameter (Teil 21 von 33)

| Name des Parameters | Bedeutung des Parameters | Standard-wert (Typ/Size) | Wahl-Wert |
|---|---|---|---|
| NBRUNWT<br>+ | Legt die Zeit in Sekunden fest, die nach ASTOP-Kommandos aus RUN-Sequenzen auf AGOGO-Kommandos gewartet wird. | 180<br>(A/4) | 10 ... 255 |
| NBSILND | Legt fest, ob in einem Startup-Modus ≠ DIALOG auf die Ausgabe der über Routing-Codes verteilten Meldungen an allen Konsolen bis „System Ready“ verzichtet wird.<br>Ausnahme: Fragen<br>Die unterdrückten Meldungen werden in der Datei SYS.CONSLOG protokolliert.<br><br>N Die Ausgabe dieser Meldungen soll erfolgen.<br>Y Auf die Ausgabe dieser Meldungen wird verzichtet. | N<br>(C/1) | N/Y |
| NRTKILL<br>* | Legt fest, ob das Operator-Kommando FORCE-JOB-CANCEL verboten oder zugelassen ist. Die Systemverwaltung kann dieses Kommando immer eingeben, unabhängig vom eingestellten Wert.<br><br>Y Kommando ist verboten, Meldung `EXC0705` wird ausgegeben<br>N Kommando ist zugelassen | Y<br>(C/1) | Y/N |
| NTPSHIGH | Legt fest, wie der Time-Server „NETWORK-TIME-PROTOCOL“ (NTP) behandelt wird, wenn in BS2000/OSD keine Information über seine Qualität (*stratum*) vorliegt.<br><br>Y Liegt kein *stratum* vor, wird dies behandelt wie ein *stratum* ≤ 4, d.h. NTP wird die höchstpriore Zeitquelle.<br><br>N (Standardwert)<br>Liegt kein *stratum* vor, wird dies behandelt wie *stratum* > 4, d.h. NTP behält seine reguläre, niedrige Priorität.<br><br>*Hinweis*<br>Dieser Parameter ist obsolet.<br>Das aktuelle BS2000-NTP liefert stets ein *stratum*. | N<br>(C/1) | Y/N |

Tabelle 55: Systemparameter (Teil 22 von 33)

| Name des Parameters | Bedeutung des Parameters | Standard-wert (Typ/Size) | Wahl-Wert |
|---|---|---|---|
| PWACTIVE | Maximale Anzahl der Kennworte in der Kennwortliste eines Auftrags (wird getrennt nach Datei- und Crypto-Kennwörtern ausgewertet).<br><br>Wird der Grenzwert erreicht, wird die Meldung DMS0691 ausgegeben und es kann für die laufende Task kein weiteres Datei- oder Crypto-Kennwort mehr eingegeben werden, bevor nicht ein anderes aus der jeweiligen Kennworttabelle ausgetragen wurde. | 15728639<br>(A/4) | 0...<br>15728639 |
| PWENTERD | Maximal erlaubte Anzahl der ADD-PASSWORD- und ADD-CRYPTO-PASSWORD-Kommandos je Task (wird getrennt nach Datei- und Crypto-Kennwörtern ausgewertet).<br><br>Wird der Grenzwert erreicht, wird die Meldung DMS0692 ausgegeben und es kann für die laufende Task kein weiteres Datei- oder Crypto-Kennwort eingegeben werden. | 2147483647<br>(A/4) | 0...<br>2147483647 |
| PWERRORS | Maximal erlaubte Anzahl der Fehlversuche bei der Kennwortüberprüfung je Task (wird getrennt nach Datei- und Crypto-Kennwörtern ausgewertet).<br><br>Wird der Grenzwert erreicht, wird ein SAT-Eintrag geschrieben bzw. an der Konsole die Meldung DMS0693 ausgegeben, wenn SECOS nicht installiert ist. | 15728639<br>(A/4) | 0...<br>St.-Wert |
| PWPENTI | Es kann festgelegt werden, ob bei nicht erfolgreicher Kennwortprüfung eine Zeitstrafe vergeben wird, indem die Dauer der Zeitstrafe in Sekunden angegeben wird. PWPENTI wird bei der Überprüfung von Datei-, JV- und Crypto-Kennwörtern herangezogen (bei Dateizugriffen und bei Crypto-Kennwörtern ist auch die Anzahl der Fehlversuche relevant). Folgende Zeitstrafen sind bei nicht erfolgreicher Kennwortprüfung vorgesehen:<br><br>Crypto: (PWPENTI) * Anzahl Fehlversuche (in Sekunden)<br>Dateien: (PWPENTI) * Anzahl Fehlversuche (in Sekunden)<br>JVs: (PWPENTI) (in Sekunden)<br><br>Ist der Standardwert 0 angegeben, wird keine Zeitstrafe verhängt. | 0<br>(A/1) | 0 - 60 |
| RDTESTPR<br>+, * | Begrenzt systemglobal die Lese-Testprivilegierung. Neben den benutzerkennungs-spezifischen Testwerten ist dieser Maximalwert einzuhalten. | 9<br>(A/1) | 0 - 9 |

Tabelle 55: Systemparameter (Teil 23 von 33)

| Name des Parameters | Bedeutung des Parameters | Standard-wert (Typ/Size) | Wahl-Wert |
|---|---|---|---|
| RECONUC | Gibt die Voreinstellung der Systembetreuung für die Rekonstruktionsanforderung an den Benutzerkatalog an.<br>Bei FAST- oder AUTOMATIC-Startup wird die entsprechende Rekonstruktion automatisch durchgeführt, d.h. ohne Eingriff des Operators.<br>Bei DIALOG-Startup kann die Voreinstellung durch den Operator verändert werden (Meldung `NSI6010`).<br>Welche Rekonstruktion durchgeführt wurde, kann unter TSOS nach „System Ready“ mit dem Kommando SHOW-SYSTEM-PARAMETERS PAR=RECONUC abgefragt werden.<br><br>N Der Benutzerkatalog bleibt unverändert.<br>B Der Benutzerkatalog wird rekonstruiert.<br>Ein neuer Benutzerkatalog wird anhand des Inhalts der Datei $TSOS.SYSSRPM.BACKUP erstellt.<br>T Der Benutzerkatalog wird rekonstruiert.<br>Ein neuer Benutzerkatalog wird erzeugt, in den ausschließlich die Benutzer eingetragen werden, die auch in der Datei $TSOS.TSOSCAT verzeichnet sind.<br>Sofern für diese Benutzer außerdem Benutzerattribute in der Datei $TSOS.SYSSRPM.BACKUP eingetragen sind, werden sie in die neuen Einträge übernommen.<br>A Der Benutzerkatalog wird rekonstruiert.<br>Ein neuer Benutzerkatalog wird anhand des Inhalts der Dateien $TSOS.SYSSRPM.BACKUP und $TSOS.TSOSCAT aufgebaut. Er enthält also Einträge für alle Benutzer, die in mind. einer der beiden Dateien verzeichnet sind. Sofern vorhanden, werden die Benutzerattribute der Datei $TSOS.SYSSRPM.BACKUP entnommen, andernfalls wird ein Standard-Benutzereintrag erstellt und die betroffenen Kennungen gesperrt.<br>R Der Benutzerkatalog wird zurückgesetzt.<br>Ein neuer Benutzerkatalog wird ausschließlich aus den Benutzern aufgebaut, die in der Datei $TSOS.TSOSCAT eingetragen sind. Alle Benutzer erhalten Standardattribute, alle Kennungen außer TSOS werden gesperrt. | N<br>(C/1) | N/B/T/A/R |

Tabelle 55: Systemparameter (Teil 24 von 33)

| **Name des Parameters** | **Bedeutung des Parameters** | **Standard-wert (Typ/Size)** | **Wahl-Wert** |
|---|---|---|---|
| SECSTART<br>+ | Schaltet den sicheren Systemstart ein oder aus. Bei eingeschaltetem sicheren Systemstart werden die Systemlauf-spezifischen Logging-Dateien $SYSAUDIT.SYS.REPLOG... der vorhergehenden Systemläufe nicht automatisch gelöscht.<br>Bei Überlauf des Meldungspuffers wird die Systemeinleitung abgebrochen. Ausgaben auf Konsolen werden, wenn sie aus Systemkomponenten erfolgen, mit dem Zeichen # vor der Uhrzeit gekennzeichnet. | N<br>(C/1) | N/Y |
| SECSTENF<br>+ | Die Systemeinleitung wird abgebrochen, wenn REPs nicht vollständig protokolliert werden können.<br>(SECSTENF: SECSTART ENFORCEMENT) | N<br>(C/1) | N/Y |
| SHUTARCH<br>+, * | Nach Eingabe des Kommandos SHUTDOWN soll geprüft werden, ob das Subsystem ARCHIVE noch benutzt wird.<br><br>N Prüfung wird nicht durchgeführt, Shutdown wird sofort eingeleitet (Standard).<br>Y Prüfung findet statt; wird ARCHIVE noch benutzt, wird der Operator gefragt, ob Shutdown trotzdem durchgeführt werden soll.<br><br>Beim VM2000-Kommando /SHUTDOWN-VM wird dieser Parameter ignoriert. Es gilt SHUTARCH=N für das Monitorsystem und alle Gastsysteme. | N<br>(C/1) | Y/N |

Tabelle 55: Systemparameter (Teil 25 von 33)

| Name des Parameters | Bedeutung des Parameters | Standard-wert (Typ/Size) | Wahl-Wert |
|---|---|---|---|
| SHUTCTL<br>* | Legt die maximale Wartezeit in Minuten fest, die nach Eingabe des Kommandos SHUTDOWN auf die Beendigung von Benutzer-Jobs gewartet werden soll. Jobs, die innerhalb dieser Wartezeit nicht beendet werden können, werden mit dem Kommando FORCE-JOB-CANCEL beendet.<br><br>0 Die Wartezeit beträgt maximal 30 Sekunden.<br>n Die Wartezeit beträgt maximal n Minuten (1 ≤ n ≤ 60).<br>Die tatsächlich benötigte Zeit zur Beendigung der Jobs ist system-spezifisch und von folgenden Faktoren abhängig:<br>– Hardware-Konfiguration (z.B. Anzahl und Leistungsfähigkeit der CPUs und IOPs, Leistungsfähigkeit der Peripherie und des Verbindungssystems)<br>– Software-Konfiguration (z.B. CPU-Quota der Gastsysteme bei VM2000-Betrieb, aktive Subsysteme zum Shutdown-Zeitpunkt, Einsatz von HIPLEX MSCF und Anzahl der Cluster)<br>– Kundenspezifische Faktoren (z.B. Anzahl und Kontext der Benutzertasks zum Shutdown-Zeitpunkt) | 0<br>(A/1) | 0...60 |
| SHUTPROC<br>+, * | Legt die Vorgehensweise bei einem Shutdown auf Anforderung, derzeit durch X2000 auf SQ-Servern, fest. Siehe „Systembeendigung über X2000 auf SQ-Servern“ auf Seite 71<br><br>BY-SHUTDOWN-CMD<br>Die Systembeendigung wird sofort eingeleitet.<br>BY-ENTER-JOB<br>Die Systembeendigung erfolgt durch einen Enter-Job. | BY-SHUTDOWN-CMD<br>(C/15) | BY-SHUT-DOWN-CMD/<br>BY-ENTER-JOB |
| SNAPTIME<br>* | Steuert die Dauer der Unterbrechung von BS2000/OSD durch die SNAP-Dump-Funktion (in Sekunden). | 24<br>(A/2) | 8...24 |
| SSMAPRI<br>* | Primärzuweisung in PAM-Blöcken für die Systemdateien<br><br>S.OUT.tsn.yyyy-mm-dd.hhmmss.cnt<br>S.LST.tsn.yyyy-mm-dd.hhmmss.cnt<br><br>und für umgewiesene Systemdateien, falls die betroffene Datei durch ein ASSIGN-<sysfile>-Kommando erzeugt wurde.<br>cnt: Zähler, 4-stellig | 24<br>(A/4) | 3...65535 |

Tabelle 55: Systemparameter (Teil 26 von 33)

| Name des Parameters | Bedeutung des Parameters | Standard-wert (Typ/Size) | Wahl-Wert |
|---|---|---|---|
| SSMASEC<br>* | Sekundärzuweisung in PAM-Blöcken für die Systemdateien<br><br>S.OUT.tsn.yyyy-mm-dd.hhmmss.cnt<br>S.LST.tsn.yyyy-mm-dd.hhmmss.cnt<br><br>und für umgewiesene Systemdateien, falls die betroffene Datei durch ein ASSIGN-<sysfile>-Kommando erzeugt wurde.<br>cnt: Zähler, 4-stellig | 24<br>(A/4) | 3...65535 |
| SSMCOPT<br>* | Bei den Kommandos kann das Fortsetzungszeichen „-“ (Bindestrich) in den Spalten 2 bis 72 angegeben werden.<br><br>Y Das Fortsetzungszeichen kann in den Spalten 2 bis 72 angegeben werden.<br>N Das Fortsetzungszeichen muss in Spalte 72 angegeben werden. | N<br>(C/1) | Y/N |
| SSMLGOF1<br>+ | Legt fest, wie SPOOLOUT-Jobs von SYSOUT- oder SYSLST-Dateien bei LOGOFF abhängig vom Zustand des SPOOL behandelt werden.<br><br>REQ-SPOOL<br>SPOOLOUT-Jobs von Systemdateien werden immer angenommen. Ist SPOOL nicht geladen, warten die Jobs bis SPOOL READY.<br>NO-SPOOL<br>Im Batchbetrieb wird LOGOFF NOSPOOL immer simuliert, d.h. es werden keine SPOOLOUT-Jobs angenommen. Im Dialogbetrieb wird der Wert von SSMLGOF2 geprüft. Wenn eine Meldung ausgegeben wird, kann sie beantwortet werden. Sonst wird LOGOFF NOSPOOL simuliert.<br>STA-SPOOL<br>Wenn SPOOL geladen ist, werden die SPOOLOUT-Aufträge angenommen. Ist SPOOL nicht geladen, wird LOGOFF NOSPOOL simuliert. | REQ-SPOOL<br>(C/9) | REQ-SPOOL/<br>NO-SPOOL/<br>STA-SPOOL |

Tabelle 55: Systemparameter (Teil 27 von 33)

| Name des Parameters | Bedeutung des Parameters | Standard-wert (Typ/Size) | Wahl-Wert |
|---|---|---|---|
| SSMLGOF2<br>+ | Legt fest, welche Meldungen der Dialog-Teilnehmer bei LOGOFF zur Ausgabe von SYSOUT- oder SYSLST-Dateien erhält.<br><br>NO<br>Mit den SPOOLOUT-Jobs wird verfahren, wie mit SSMLGOF1 festgelegt. Es wird keine Meldung ausgegeben.<br>YES<br>Der Dialogteilnehmer erhält je nach Zustand von SPOOL folgende Meldungen:<br>SPOOL nicht geladen: SSM2066<br>Der Dialogteilnehmer kann angeben, ob er warten will, bis SPOOL geladen ist, oder ob die SPOOLOUT-Anforderung ignoriert werden soll.<br>SPOOL ist geladen: SSM2075<br>Der Dialogteilnehmer kann angeben, ob der SPOOLOUT-Job durchgeführt oder ignoriert werden soll.<br>SPOOL<br>Wenn SPOOL geladen ist, erhält der Dialogteilnehmer keine Meldung. Ist SPOOL nicht geladen, erhält er die Meldung SSM2066 und kann angeben, ob er warten will, bis SPOOL geladen wird, oder ob die SPOOLOUT-Anforderung ignoriert werden soll. | NO<br>(C/5) | NO/YES/<br>SPOOL |
| SSMMILOG<br>* | Legt fest, ob bei der Zuweisung von SYSLST bzw. SYSOUT in ein PLAM-Bibliothekselement die Ausgabesätze zusätzlich in eine Datei geschrieben werden.<br><br>Y Bei jeder Zuweisung von SYSLST bzw. SYSOUT in ein PLAM-Bibliothekselement wird eine Datei als Kopie angelegt und jeder Ausgabesatz wird zusätzlich in diese Datei geschrieben. Die Kopie erhält folgenden Namen:<br>S.LST.lib.elem.tsn.counter bzw.<br>S.OUT.lib.elem.tsn.counter<br>lib: die ersten 7 Zeichen der PLAM-Bibliothek<br>elem: die ersten 7 Zeichen des Elements<br>tsn : TSN der Task, 4 Zeichen<br>counter: Zähler, 4-stellig<br>N Ausgabesätze werden nicht zusätzlich kopiert. | N<br>(C/1) | Y/N |

Tabelle 55: Systemparameter (Teil 28 von 33)

| Name des Parameters | Bedeutung des Parameters | Standard-wert (Typ/Size) | Wahl-Wert |
|---|---|---|---|
| SSMOUT<br>+, * | Legt den Standardwert der Systemausgabe für die Kommandos ENTER-PROCEDURE, EXIT-JOB, LOGOFF und CANCEL-JOB fest. In einem Rechnerverbund wird SSMOUT am Zielrechner ausgewertet.<br><br>PRINT<br>Systemausgaben werden mit SPOOL ausgegeben.<br>MAIL<br>Systemausgaben werden per E-Mail an die im Benutzereintrag der Benutzerkennung eingetragenen Empfängeradressen (EMAIL-ADDRESS) gesandt. Ist kein Versand möglich, wird wie bei PRINT verfahren. | PRINT<br>(C/5) | PRINT/<br>MAIL |
| SSMSDEVB<br>* | Legt den Gerätetyp beim Wechsel von Platte zu Band für einen Batch-Task fest. | TAPE<br>(C/8) | Installa-tions-abhängig |
| SSMSDEVD<br>* | Legt den Gerätetyp beim Wechsel von Platte zu Band fest, wenn auf die Meldung SSM2035 (Plattenspeichermangel) mit 'M=mbk' geantwortet wird. | TAPE-C1<br>(C/8) | Installa-tions-abhängig |
| STUPTYPE | Gibt die Voreinstellung der Systembetreuung für die Art des Systemstarts an.<br><br>Bei FAST- oder AUTOMATIC-Startup wird der entsprechende Systemstart automatisch durchgeführt, d.h. ohne Eingriff des Operators.<br><br>Bei DIALOG-Startup kann die Voreinstellung durch den Operator verändert werden (Meldung NSI6005).<br>Welcher Systemstart durchgeführt wurde, kann mit dem Kommando SHOW-SYSTEM-PARAMETERS PAR=STUPTYPE abgefragt werden.<br><br>W Warmstart<br>C Kaltstart<br>J First-Start mit Rücksetzen des Benutzerkatalogs<br>Es werden der Benutzerkatalog, die JMS-Auftragswarteschlange und die TSN-Datei zurückgesetzt.<br>S Selective-Start<br>T First-Start mit Rücksetzen der TSN-Datei<br>Es werden die JMS-Auftragswarteschlange und die TSN-Datei zurückgesetzt.<br>Z Zip-Start | W<br>(C/1) | W/C/J/S/<br>T/Z |

Tabelle 55: Systemparameter (Teil 29 von 33)

| Name des Parameters | Bedeutung des Parameters | Standard-wert (Typ/Size) | Wahl-Wert |
|---|---|---|---|
| SVC79<br>+ | Es kann die Anwendbarkeit des SVC79 (Übergang vom Benutzerzustand in den privilegierten Systemzustand) eingeschränkt werden.<br><br>0 SVC79 darf nur von Tasks ausgeführt werden, die unter der Benutzerkennung TSOS oder SERVICE (mit ETCBWOPT) ablaufen. Beim ersten SVC79 einer Task wird eine entsprechende Meldung an der Bedienstation ausgegeben.<br>1 Der Operator muss beim ersten SVC79 einer Task den Übergang bestätigen. Zusätzlich gilt, dass nur TSOS-Tasks und SERVICE-Tasks diesen SVC abgeben dürfen.<br>2 SVC79 ist nur für Tasks zulässig, die unter der Benutzerkennung SERVICE arbeiten. Zusätzlich muss der Operator den Übergang bestätigen.<br>3 Es gibt keine Möglichkeit, vom Benutzerzustand in den privilegierten Systemzustand zu wechseln. | 0<br>(A/4) | 0 - 3 |
| SYSGBSIZ | Puffergröße für Datenstationen (in Byte).<br><br>*Hinweis*<br>Die Puffergröße kann im SETBF-Makro für jede Datenstation individuell bestimmt werden (siehe Handbuch „Makroaufrufe an den Ablaufteil“ [30]). | 256<br>(A/4) | 256..2048 |
| SYSGJASL<br>* | Standardwert für den Operanden ADDRESS-SPACE-LIMIT im Kommando ADD-USER (Benutzerspeicher für eine Task unter dieser Benutzerkennung). | 16<br>(A/4) | 1...<br>2147483647 |
| SYSGJCLA | Standardwert für den Operanden SPOOLOUT-CLASS im Kommando ADD-USER (SPOOLOUT-Klasse).<br><br>0 Keine SPOOLOUT-Klasse für diese Benutzerkennung/Abrechnungsnummer.<br>1...255<br>Die angegebene SPOOLOUT-Klasse wird der Benutzerkennung/Abrechnungsnummer zugeordnet. | 0<br>(A/2) | 0...255 |
| SYSGJCPU<br>* | Standardwert für den Operanden CPU-LIMIT im Kommando ADD-USER (gesamte Laufzeit für eine Benutzerkennung/Abrechnungsnummer). | 65535<br>(A/4) | 0...<br>2147483647 |
| SYSGJPRI<br>* | Standardwert für den Operanden MAXIMUM-RUN-PRIORITY im Kommando ADD-USER. | 255<br>(A/2) | 1...255 |

Tabelle 55: Systemparameter (Teil 30 von 33)

| Name des Parameters | Bedeutung des Parameters | Standard-wert (Typ/Size) | Wahl-Wert |
|---|---|---|---|
| SYSTHOOK | Schränkt die Verfügbarkeit von Systemhook-Kommandos ein. Systemhook ist ein internes Diagnosetool.<br><br>SYSTHOOK=YES<br>alle Systemhook-Kommandos sind erlaubt.<br>SYSTHOOK=NO<br>die Systemhook-Kommandos ADD-SYSTEM-HOOK und MODIFY-SYSTEM-HOOK-PROCESSING sind verboten; alle anderen sind erlaubt. | NO (C/3) | NO/YES |
| SYSVECT | Größe des Systemvektors in Seiten (globaler Speicherbereich für System-Exit-Routinen). | 0<br>(A/2) | 0...15 |
| TASKVECT | Größe des Taskvektors in Seiten (lokaler Speicherbereich für System-Exit-Routinen). | 0<br>(A/2) | 0...15 |
| TCHOFLO<br>* | Legt den Standardwert für die Art der Überlaufkontrolle an Datenstationen fest (entspricht dem Kommando MODIFY-TERMINAL-OPTIONS, siehe Handbuch „Kommandos“ [27])<br><br>USER-ACKNOWLEDGE<br>Überlaufkontrolle durch Quittung des Benutzers.<br>TIMEOUT<br>Überlaufkontrolle durch Zeitverzögerung von 6 Sek.<br>NO-CONTROL<br>keine Überlaufkontrolle. | ACK<br>(C/5) | NO/<br>TIMER/<br>ACK |
| TCHREAD<br>* | Legt den Standardwert für die Betriebsart der Datenstationen 816x und 975x fest; entspricht dem Kommando MODIFY-TERMINAL-OPTIONS ..,WRITE-READ-MODE= ... (siehe Handbuch „Kommandos“ [27]).<br><br>MODIFIED-FIELDS (=MODIF)<br>Lesen modifizierter Felder.<br>NO-FIELDS (=UNPRO)<br>Lesen ungeschützter Felder. | MODIF<br>(C/5) | MODIF/<br>UNPRO |
| TCHTACK<br>* | Legt den Standardwert für das Quittungsverfahren bei Ausgaben an Datenstationen fest; entspricht dem Kommando MODIFY-TERMINAL-OPTIONS ...,ACKNOWLEDGE-OUTPUT=... (siehe Handbuch „Kommandos“ [27]).<br><br>YES Quittungsverfahren eingeschaltet.<br>NO Quittungsverfahren ausgeschaltet (wird bei 8103 nicht ausgewertet). | NO<br>(C/3) | NO/YES |

Tabelle 55: Systemparameter (Teil 31 von 33)

| **Name des Parameters** | **Bedeutung des Parameters** | **Standard-wert (Typ/Size)** | **Wahl-Wert** |
|---|---|---|---|
| TEMPFILE<br>+ | Sonderzeichen, das als Präfix bei temporären Dateien und Jobvariablen vor den Datei- bzw. JV-Namen gesetzt werden muss.<br><br>NO Das System arbeitet ohne temporäre Dateien bzw. Jobvariablen.<br>sonderzeichen (# oder @)<br>Dateien bzw. JV, die mit dem vorangestellten Sonderzeichen erzeugt werden, sind temporär und erhalten den Namen:<br>S.sysid.nnnn.dateiname<br>bzw.<br>S.sysid.nnnn.jvname<br>(sysid = SYS-ID des Home-Pubsets<br>nnnn = TSN des erzeugenden Jobs).<br>Sie werden bei LOGOFF automatisch gelöscht.<br><br>*Hinweis*<br>Die temporären Dateinamen können sich in einer späteren BS2000-Version ändern, sie stellen keine Systemschnittstelle dar. | C'NO'<br>(C/2) | NO/<br>C'#'/'@' |
| VMGIORAL<br>* | Legt fest, ob ein VM2000-globaler Wegschaltauftrag (Kommando DETACH-DEVICE oder REMOVE-DEVICE-CONNECTION mit SCOPE=*VM2000-GLOBAL) im Gastsystem ausgeführt werden darf.<br><br>YES Wegschaltauftrag darf ausgeführt werden.<br>NO Wegschaltauftrag darf nicht ausgeführt werden | YES | YES/NO |
| VSEQPRI | Beim Anfordern von Bändern hat die Angabe des Dateinamens bzw. der Dateiabschnittsnummer Vorrang gegenüber der Angabe einer Archivnummer (VSN).<br><br>1 Angabe Dateiname/Dateiabschnittsnummer hat Vorrang.<br>Wenn VSEQPRI den Wert 1 hat, kann<br>- beim Importieren einer Banddatei die VSN weggelassen werden (FILE-Makro mit STATE=FOREIGN)<br>- beim Bearbeiten einer katalogisierten Banddatei eine andere VSN zugewiesen werden als im Dateikatalog eingetragen (z.B. VSN der Sicherungskopie).<br>0 Angabe Archivnummer hat Vorrang. | 0<br>(C/1) | 0/1 |

Tabelle 55: Systemparameter (Teil 32 von 33)

| Name des Parameters | Bedeutung des Parameters | Standard-wert (Typ/Size) | Wahl-Wert |
|---|---|---|---|
| WRTESTPR<br>+, * | Begrenzt systemglobal die Schreib-Testprivilegierung. Neben den Benutzerkennungs-spezifischen Testwerten ist dieser Maximalwert einzuhalten. | 9<br>(A/1) | 0 - 9 |

Tabelle 55: Systemparameter (Teil 33 von 33)

## 19.2 Zeichensatz bei Ein-/Ausgabe über Bedienstation

Bei **Eingaben** über Bedienstation werden alle auf der jeweiligen Tastatur eingebbaren Zeichen unterstützt. Sie werden intern vom bedienstationeigenen Zeichencode in EDCDIC.DF.04 umgewandelt.

Bei **Ausgaben** über Bedienstation wird der Zeichencode umgesetzt in einen Zeichencode, der auf der jeweiligen Bedienstation darstellbar ist. Alle nichtabdruckbaren Zeichen, außer X'15', werden in Leerzeichen X'40' umgewandelt.

Die folgende Umsetztabelle zeigt, welche Zeichen an Bedienstationen 3020-3027 unterstützt werden:

<table>
<tr><th></th><th>0</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>8</th><th>9</th><th>A</th><th>B</th><th>C</th><th>D</th><th>E</th><th>F</th></tr>
<tr><td>0</td><td></td><td></td><td></td><td></td><td>SP</td><td>&</td><td>–</td><td></td><td></td><td></td><td></td><td></td><td>{|ä</td><td>}|ü</td><td>\|Ö</td><td>0</td></tr>
<tr><td>1</td><td></td><td></td><td></td><td></td><td></td><td></td><td>/</td><td></td><td>a</td><td>j</td><td>~|ß</td><td></td><td>A</td><td>J</td><td></td><td>1</td></tr>
<tr><td>2</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>b</td><td>k</td><td>s</td><td></td><td>B</td><td>K</td><td>S</td><td>2</td></tr>
<tr><td>3</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>c</td><td>l</td><td>t</td><td></td><td>C</td><td>L</td><td>T</td><td>3</td></tr>
<tr><td>4</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>d</td><td>m</td><td>u</td><td></td><td>D</td><td>M</td><td>U</td><td>4</td></tr>
<tr><td>5</td><td></td><td>NL</td><td></td><td></td><td></td><td></td><td></td><td></td><td>e</td><td>n</td><td>v</td><td></td><td>E</td><td>N</td><td>V</td><td>5</td></tr>
<tr><td>6</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>f</td><td>o</td><td>w</td><td></td><td>F</td><td>O</td><td>W</td><td>6</td></tr>
<tr><td>7</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>g</td><td>p</td><td>x</td><td></td><td>G</td><td>P</td><td>X</td><td>7</td></tr>
<tr><td>8</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>h</td><td>q</td><td>y</td><td></td><td>H</td><td>Q</td><td>Y</td><td>8</td></tr>
<tr><td>9</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>`</td><td>i</td><td>r</td><td>z</td><td></td><td>I</td><td>R</td><td>Z</td><td>9</td></tr>
<tr><td>A</td><td></td><td></td><td></td><td></td><td>[|Ä</td><td>!|</td><td>^</td><td>:</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>B</td><td></td><td></td><td></td><td></td><td>·</td><td>|$</td><td>,</td><td>#|</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>C</td><td></td><td></td><td></td><td></td><td><</td><td>*</td><td>%</td><td>@|§</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>D</td><td></td><td></td><td></td><td></td><td>(</td><td>)</td><td>–</td><td>'</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>E</td><td></td><td></td><td></td><td></td><td>+</td><td>;</td><td>></td><td>=</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>F</td><td></td><td></td><td></td><td></td><td>||ö</td><td>]|Ü</td><td>?</td><td>"</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
</table>

Tabelle 56: Zeichensatz an Bedienstationen 3020-3027

Die folgende Umsetztabelle zeigt, welche Zeichen an Bedienstationen 3027-C unterstützt werden:

<table>
<tr><th></th><th>0</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>8</th><th>9</th><th>A</th><th>B</th><th>C</th><th>D</th><th>E</th><th>F</th></tr>
<tr><td>0</td><td></td><td></td><td></td><td></td><td>SP</td><td>&</td><td>–</td><td></td><td></td><td></td><td></td><td></td><td>{|ä</td><td>}|ü</td><td>\|Ö</td><td>0</td></tr>
<tr><td>1</td><td></td><td></td><td></td><td></td><td></td><td></td><td>/</td><td></td><td>a</td><td>j</td><td>~|ß</td><td></td><td>A</td><td>J</td><td></td><td>1</td></tr>
<tr><td>2</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>b</td><td>k</td><td>s</td><td></td><td>B</td><td>K</td><td>S</td><td>2</td></tr>
<tr><td>3</td><td></td><td></td><td></td><td></td><td></td><td></td><td>[|Ä</td><td></td><td>c</td><td>l</td><td>t</td><td></td><td>C</td><td>L</td><td>T</td><td>3</td></tr>
<tr><td>4</td><td></td><td></td><td></td><td></td><td></td><td></td><td>]|Ü</td><td></td><td>d</td><td>m</td><td>u</td><td></td><td>D</td><td>M</td><td>U</td><td>4</td></tr>
<tr><td>5</td><td></td><td>NL</td><td></td><td></td><td></td><td></td><td></td><td></td><td>e</td><td>n</td><td>v</td><td></td><td>E</td><td>N</td><td>V</td><td>5</td></tr>
<tr><td>6</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>f</td><td>o</td><td>w</td><td></td><td>F</td><td>O</td><td>W</td><td>6</td></tr>
<tr><td>7</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>g</td><td>p</td><td>x</td><td></td><td>G</td><td>P</td><td>X</td><td>7</td></tr>
<tr><td>8</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>h</td><td>q</td><td>y</td><td></td><td>H</td><td>Q</td><td>Y</td><td>8</td></tr>
<tr><td>9</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>`</td><td>i</td><td>r</td><td>z</td><td></td><td>I</td><td>R</td><td>Z</td><td>9</td></tr>
<tr><td>A</td><td></td><td></td><td></td><td></td><td></td><td>!|</td><td>^</td><td>:</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>B</td><td></td><td></td><td></td><td></td><td>.</td><td>|$</td><td>,</td><td>#|</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>C</td><td></td><td></td><td></td><td></td><td><</td><td>*</td><td>%</td><td>@|</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>D</td><td></td><td></td><td></td><td></td><td>(</td><td>)</td><td>–</td><td>'</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>E</td><td></td><td></td><td></td><td></td><td>+</td><td>;</td><td>></td><td>=</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>F</td><td></td><td></td><td></td><td></td><td>|ö</td><td></td><td>?</td><td>"</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
</table>

Tabelle 57: Zeichensatz an Bedienstationen 3027-C

## 19.3 Übersicht zu Testprivilegien

Die Testprivilegien steuern die Ausführbarkeit von Software- und Hardware-Diagnosetätigkeiten in BS2000/OSD. Der Hauptnutzer ist die Testhilfe AID, siehe Handbuch „AID“ [2].

Die Testprivilegien unterteilen sich in Lese- und Schreibprivilegien. Das Schreibprivileg eines Benutzers darf nicht größer sein als sein Lese-Privileg. Lese- und Schreibprivilegien sind jeweils hierarchisch definiert von 1 (niedrig) bis 9 (hoch). Eine Privilegierung mit dem Wert n umfasst implizit die Zugriffsberechtigungen der niedrigeren Privilegienstufen 1 bis n-1.

Die maximalen Testprivilegien für eine Benutzerkennung werden über die privilegierten Kommandos ADD-USER bzw. MODIFY-USER ATTRIBUTES festgelegt. Systemweite Höchstwerte für die Lese- und Schreibprivilegien können über die Systemparameter RDTESTPR und WRTESTPR (siehe Seite 755 und Seite 765) definiert werden.
Die taskspezifische Einstellung erfolgt über das Kommando MODIFY-TEST-OPTIONS.

Informationen über die aktuellen Testprivilegien liefert das Kommando SHOW-TEST-OPTIONS.

Die nachfolgenden Tabellen geben eine Übersicht über die Testprivilegien bei AID, bei weiteren Software-Diagnoseprodukten und bei der Online-Wartung für die Hardware

### 19.3.1 Testprivilegien bei AID

Wurde aus einer lesegeschützten Datei (nach)geladen und das Lesekennwort nicht angegeben, so liegt eine „Execute-Only“-Situation vor. In einer „Execute-Only-Situation“ ist, unabhängig von den eingestellten Testprivilegien, kein Testen mit AID möglich.

| Wert | Charakteristik |
|---|---|
| 1 | – Zugriff auf die Seiten im eigenen Benutzeradressraum[1)] und im Systemadressraum[2)], die keine Secret Pages[3)] und mit den Zugriffsrechten des normalen Benutzers lesbar sind<br>– Zugriff auf die Benutzer-PCBs der eigenen Task |
| 2 | – zusätzlich Zugriff auf den TCB der eigenen Task |
| 3 | – zusätzlich Zugriff auf die Secret Pages im eigenen Benutzeradressraum, die mit den Zugriffsrechten des normalen Benutzers lesbar sind |
| 4 | nicht vergeben |
| 5 | nicht vergeben |
| 6 | – zusätzlich Zugriff auf die Seiten im eigenen Benutzeradressraum, die keine Secret Pages und nur mit den Zugriffsrechten des Betriebssystems lesbar sind<br>– zusätzlich Zugriff auf die System-PCBs der eigenen Task und auf die XVT |
| 7 | nicht vergeben |
| 8 | – zusätzlich Zugriff auf alle Seiten im Systemadressraum, alle Secret Pages und alle Seiten anderer Tasks<br>– zusätzlich Zugriff auf die Benutzer-PCBs, System-PCBs und TCBs anderer Tasks |
| 9 | nicht vergeben |

Tabelle 58: Lese-Testprivilegien bei AID-Lesezugriffen

| Wert | Charakteristik |
|---|---|
| 1 | – Zugriff auf die Seiten im eigenen Benutzeradressraum[1)], die keine Secret Pages[3)] und mit den Zugriffsrechten des normalen Benutzers beschreibbar sind<br>– zusätzlich Zugriff auf die tasklokalen Seiten des Benutzers im eigenen Programmraum, die keine Secret Pages und mit den Zugriffsrechten des normalen Benutzers „nur lesbar“[4)] sind<br>– Zugriff auf die Benutzer-PCBs der eigenen Task |
| 2 | nicht vergeben |
| 3 | – zusätzlich Zugriff auf die Secret Pages im eigenen Benutzeradressraum, die mit den Zugriffsrechten des normalen Benutzers beschreibbar sind<br>– zusätzlich Zugriff auf die tasklokalen Secret Pages des Benutzers im eigenen Programmraum, die mit den Zugriffsrechten des normalen Benutzers „nur lesbar“ sind |
| 4 | nicht vergeben |

Tabelle 59: Schreib-Testprivilegien bei AID-Schreibzugriffen (Teil 1 von 2)

| Wert | Charakteristik |
|---|---|
| 5 | nicht vergeben |
| 6 | – zusätzlich Zugriff auf die Seiten im eigenen Benutzeradressraum, die keine Secret Pages und nur mit den Zugriffsrechten des Betriebssystems beschreibbar sind |
| 7 | nicht vergeben |
| 8 | – zusätzlich Zugriff auf alle Seiten im Systemadressraum[2)], alle Secret Pages und alle Seiten anderer Tasks<br>– zusätzlich Zugriff auf die XVT sowie auf die Benutzer-PCBs, System-PCBs und TCBs aller Tasks |
| 9 | nicht vergeben |

Tabelle 59: Schreib-Testprivilegien bei AID-Schreibzugriffen (Teil 2 von 2)

*Legende*

1) Programmraum- und Datenräume des Benutzers
2) Systemraum- und Datenräume des Systems
3) gegen Diagnosezugriffe geschützte Seiten
4) Das Seitenattribut „nur lesbar“ wird in diesem Fall von AID ignoriert.

### Anmerkung zum OWN-UID-DEBUGGING

Über den Operanden OWN-UID-DEBUGGING beim Kommando MODIFY-TEST-OPTIONS kann der „statische AID-Test mit niedriger Testprivilegierung“ für andere Tasks, die unter der eigenen Benutzerkennung laufen, zugelassen werden. Falls dies der Fall ist, sind bei lesenden Zugriffen auf den Benutzeradressraum dieser Tasks die gleichen Testprivilegien ausreichend wie bei Zugriffen auf den Benutzeradressraum der eigenen Task.

Für schreibende Zugriffe auf den Benutzeradressraum dieser Tasks ist mindestens ein Schreibprivileg von 2 erforderlich, darüber hinaus gelten die gleichen Testprivilegien wie bei Schreibzugriffen auf den Benutzeradressraum der eigenen Task. Bei Schreibzugriffen auf Memory-Pool-Seiten dieser Tasks wird grundsätzlich das Schreibprivileg 8 verlangt.

### 19.3.2 Testprivilegien bei weiteren Software-Diagnoseprodukten

| Wert | Produkt | Charakteristik |
|---|---|---|
| 3 | CDUMP | Ein Systemdump darf unter einer normalen Benutzerkennung gezogen werden. |
| 8 | ANITA | Das aktive System darf mit DAMP und anderen Benutzerprogrammen, die ANITA nutzen, diagnostiziert werden. |

Tabelle 60: Lese-Testprivilegien der Software-Diagnoseprodukte

### 19.3.3 Testprivilegien bei der Online-Wartung für die Hardware

| Wert | Charakteristik |
|---|---|
| 1 - 3 | Unbedenklich in Bezug auf Datenschutz und Datensicherheit. Es werden nur eigene Datenmuster verwendet. |
| 4 | Das Programm kann Daten aus dem Kundenbereich lesen. |
| 5 | wird nicht verwendet |
| 6 | Das Programm kann Daten aus dem Kundenbereich überschreiben. |
| 7 - 8 | wird nicht verwendet |
| 9 | Diese Schutzstufe hebt im Systemmodul WARTOPT die Überwachung der Ein-/Ausgaben auf Systemunschädlichkeit auf. Die Datensicherheit von Kundendaten ist nicht gewährleistet. |

Tabelle 61: Lese-Testprivilegien der Online-Wartung

# Literatur

Die Handbücher finden Sie im Internet unter *http://manuals.ts.fujitsu.com*. Handbücher, die mit einer Bestellnummer angezeigt werden, können Sie in auch gedruckter Form bestellen.


[1] **BS2000/OSD-BC**
**Abrechnungssätze**
Benutzerhandbuch

[2] **AID** (BS2000/OSD)
**Advanced Interactive Debugger**
Basishandbuch

[3] **ARCHIVE** (BS2000/OSD)
Benutzerhandbuch

[4] **openNet Server** (BS2000/OSD)
**BCAM**
Benutzerhandbuch

[5] **BLSSERV**
**Bindelader-Starter in BS2000/OSD**
Benutzerhandbuch

[6] **BINDER**
**Binder in BS2000/OSD**
Benutzerhandbuch

[7] **CALENDAR** (BS2000/OSD)
Benutzerhandbuch

[8] **ETERNUS CS HE**
**Inbetriebnahme und Administration des ETERNUS CS Highend**
Benutzerhandbuch

[9] **openCRYPT** (BS2000/OSD)
**Sicherheit mit Kryptographie**
Benutzerhandbuch

[10] **DAB** (BS2000/OSD)
**Disk Access Buffer**
Benutzerhandbuch

[11] **DCAM** (BS2000/OSD, TRANSDATA)
**COBOL-Aufrufe**
Benutzerhandbuch

[12] **DCAM** (BS2000/OSD, TRANSDATA)
**Makroaufrufe**
Benutzerhandbuch

[13] **DCAM** (BS2000/OSD, TRANSDATA)
**Programmschnittstellen**
Beschreibung

[14] **BS2000/OSD-BC**
**Diagnosehandbuch**
Benutzerhandbuch

[15] **BS2000/OSD-BC**
**Dienstprogramme**
Benutzerhandbuch

[16] **Distributed Print Services** (BS2000/OSD)
**Drucken in Computernetzen**
Benutzerhandbuch

[17] **DRV** (BS2000/OSD)
**Dual Recording by Volume**
Benutzerhandbuch

[18] **DSSM**
**Verwaltung von Subsystemen in BS2000/OSD**
Benutzerhandbuch

[19] **BS2000/OSD-BC**
**Einführung in das DVS**
Benutzerhandbuch

[20] **BS2000/OSD-BC**
**DVS-Makros**
Benutzerhandbuch

[21] **ELSA** (BS2000/OSD)
**Error Logging System Analysis**
Benutzerhandbuch

[22] **FDDRL** (BS2000/OSD)
Benutzerhandbuch

[23] **openFT für BS2000/OSD**
**Enterprise File Transfer in der offenen Welt**
Benutzerhandbuch

[24] **HSMS** (BS2000/OSD)
**Hierarchisches Speicher Management System**
Benutzerhandbuch

[25] **IMON** (BS2000/OSD)
**Installationsmonitor**
Benutzerhandbuch

[26] **JV** (BS2000/OSD)
**Jobvariablen**
Benutzerhandbuch

[27] **BS2000/OSD-BC**
**Kommandos**
Benutzerhandbuch

[28] **BS2000/OSD-BC**
**System Managed Storage**
Benutzerhandbuch

[29] **LMS** (BS2000/OSD)
**SDF-Format**
Benutzerhandbuch

[30] **BS2000/OSD-BC**
**Makroaufrufe an den Ablaufteil**
Benutzerhandbuch

[31] **MAREN** (BS2000/OSD)
**Band 1: Grundlagen der MBK-Verwaltung**
**Band 2: Benutzerschnittstellen**
Benutzerhandbuch

[32] **BS2000/OSD-BC**
**Migration Guide**
Benutzerhandbuch

[33] **HIPLEX MSCF** (BS2000/OSD)
**BS2000-Rechner im Verbund**
Benutzerhandbuch

[34] **OMNIS/OMNIS-MENU** (TRANSDATA, BS2000/OSD)
**Funktionen und Kommandos**
Benutzerhandbuch

[35] **OMNIS/OMNIS-MENU** (TRANSDATA, BS2000/OSD)
**Administration und Programmierung**
Benutzerhandbuch

[36] **OMNIS-PROP** (TRANSDATA, BS2000/OSD)
Benutzerhandbuch

[37] **BS2000/OSD-BC**
**Operator-Kommandos (ISP-Format)**
Benutzerhandbuch

[38] **PCS** (BS2000/OSD)
**Performance Control Subsystem**
Benutzerhandbuch

[39] **BS2000/OSD-BC**
**Performance Handbuch**
Benutzerhandbuch

[40] **POSIX** (BS2000/OSD)
**Kommandos**
Benutzerhandbuch

[41] **POSIX** (BS2000/OSD)
**Grundlagen für Anwender und Systemverwalter**
Benutzerhandbuch

[42] **PROP-XT** (BS2000/OSD)
**Programmiertes Operating mit komfortablen Sprachmitteln von SDF-P**
Produkthandbuch

[43] **ROBAR** (BS2000/OSD, Linux)
**Steuerung von MBK-Archivsystemen**
Benutzerhandbuch

[44] **PRM** (BS2000/OSD)
Benutzerhandbuch

[45] **SDF** (BS2000/OSD)
**Dialogschnittstelle SDF**
Benutzerhandbuch

[46] **SDF-A** (BS2000/OSD)
Benutzerhandbuch

[47] **SDF-P** (BS2000/OSD)
**Programmieren in der Kommandosprache**
Benutzerhandbuch

[48] **SECOS** (BS2000/OSD)
**Security Control System - Zugangs- und Zugriffskontrolle**
Benutzerhandbuch

[49] **SECOS** (BS2000/OSD)
**Security Control System - Beweissicherung**
Benutzerhandbuch

[50] **SHC-OSD / SCCA-BS2** (BS2000/OSD)
**Storage Management für BS2000/OSD**
Benutzerhandbuch

[51] **openSM2** (BS2000/OSD)
**Software Monitor**
Benutzerhandbuch

[52] **SM2-PA** (BS2000/OSD)
**SM2-Programmanalysator**
Benutzerhandbuch

[53] **SNS** (BS2000/OSD)
**SPOOL Notification Service**
Benutzerhandbuch

[54] **SPACEOPT** (BS2000/OSD)
**Optimierung und Reorganisation von Platten**
Benutzerhandbuch

[55] **SPOOL** (BS2000/OSD)
Benutzerhandbuch

[56] **SPSERVE** (BS2000/OSD)
Benutzerhandbuch

[57] **BS2000/OSD-BC**
**Systeminstallation**
Benutzerhandbuch

[58] **BS2000/OSD-BC**
**System-Exits**
Benutzerhandbuch

[59] **Business Server der SQ-Serie**
**Bedienen und Verwalten**
Benutzerhandbuch

[60] **Unicode im BS2000/OSD**
Übersichtshandbuch

[61] **openUTM** (BS2000/OSD, UNIX, Windows)
**Anwendungen generieren**
Benutzerhandbuch

[62] **VM2000** (BS2000/OSD)
**Virtuelles Maschinensystem**
Benutzerhandbuch

# Stichwörter

## A

## Z